



UNIVERSIDAD  
POLITECNICA  
DE VALENCIA



Proyecto Final de Carrera

# **Sistema basado en dispositivos móviles para la adquisición de datos de consumo en contadores de agua**

Abril de 2007, Valencia

*Francisco Javier Cano Muñoz*  
[fracamuo@inf.upv.es](mailto:fracamuo@inf.upv.es)

**Dirigido por :**  
*Joan Fons i Cors*  
[jjfons@dsic.upv.es](mailto:jjfons@dsic.upv.es)

**Proyecto :**  
II-B-DSIC- 205/06



## **Agradecimientos**

A mi familia, por descontado. Aún no sé como voy a pagarles por todo el apoyo y educación que he recibido de ellos.

A l@s amig@s; cada uno ha dado su martillazo para forjar lo que hoy soy.

A todos los personajes, ilustres o no, que han hecho de la informática lo que es hoy.

A tí por, al menos, echarle un vistazo a este trabajo :)





# Índice

<b>CAPÍTULO 1- INTRODUCCIÓN.....</b>	<b>1</b>
1.1.- <i>Aguas del Bullent, S.A.</i> .....	1
1.2.- <i>Descripción del documento</i> .....	1
<b>CAPÍTULO 2- CONTEXTO Y TECNOLOGÍA.....</b>	<b>3</b>
2.1.- <i>Estructura informática existente</i> .....	3
2.1.1.- <i>Cliente servidor</i> .....	4
2.2.- <i>Proceso de toma de lecturas</i> .....	4
2.3.- <i>Propuestas</i> .....	5
2.3.1.- <i>Dispositivos móviles</i> .....	7
2.3.2.- <i>Sistema actual</i> .....	8
2.3.3.- <i>Sincronización</i> .....	9
<b>CAPÍTULO 3- DISPOSITIVOS MÓVILES.....</b>	<b>11</b>
3.1.- <i>Dispositivos</i> .....	11
3.1.1.- <i>Datalogic Blackjet</i> .....	11
3.1.2.- <i>Limitaciones y ventajas del dispositivo</i> .....	14
3.2.- <i>Diseño del sistema</i> .....	15
3.2.1.- <i>Nivel de datos</i> .....	15
3.2.2.- <i>Lógica de negocio</i> .....	18
3.2.3.- <i>Interfaz de usuario</i> .....	21
3.2.4.- <i>Gestores de la aplicación</i> .....	32
3.3.- <i>Entorno de desarrollo</i> .....	40
<b>CAPÍTULO 4- APLICACIÓN OLIVANOVA.....</b>	<b>41</b>
4.1.- <i>Nuevas funcionalidades añadidas</i> .....	41
4.1.1.- <i>Sincronización</i> .....	42
4.1.2.- <i>Impresión</i> .....	44
<b>CAPÍTULO 5- SINCRONIZACIÓN.....</b>	<b>47</b>
5.1.- <i>Protocolo de sincronización</i> .....	48
5.1.1.- <i>Protocolo en dispositivo</i> .....	48
5.1.2.- <i>Protocolo en AguasCli</i> .....	49
5.1.3.- <i>Operaciones del protocolo</i> .....	50
5.1.4.- <i>Discriminación de los cambios</i> .....	52
5.1.5.- <i>Envío de datos</i> .....	53
5.2.- <i>AguasCli como intermediario</i> .....	54
5.2.1.- <i>Uso de la metodología OlivaNova</i> .....	54
5.2.2.- <i>Control de la carga del proceso</i> .....	55
5.3.- <i>Errores y cancelaciones</i> .....	56
5.3.1.- <i>Errores</i> .....	56
5.3.2.- <i>Cancelaciones</i> .....	57
<b>CAPÍTULO 6- CÓDIGOS DE BARRAS.....</b>	<b>59</b>
6.1.- <i>Lector de códigos de barras del dispositivo</i> .....	59
6.2.- <i>Códigos de barras existentes</i> .....	59
6.2.1.- <i>Formato de un código de barras</i> .....	60

6.2.2.- Clasificaciones de códigos de barras.....	60
6.2.3.- Estándares más utilizados.....	61
<b>6.3.- Código de barras elegido.....</b>	<b>63</b>
<b>6.4.- Impresión.....</b>	<b>64</b>
6.4.1.- Nueva funcionalidad.....	64
6.4.2.- Impresora.....	65
6.4.3.- Soporte físico.....	66
6.4.4.- Impresión de todas las rutas.....	67
<b>CAPÍTULO 7- IMPLANTACIÓN.....</b>	<b>69</b>
<b>7.1.- Primera etapa, sugerencias, cambios, mejoras.....</b>	<b>69</b>
7.1.1.- Visita a Aguas del Bullent S.A.....	69
7.1.2.- Personal técnico.....	70
7.1.3.- Personal de oficina.....	71
<b>7.2.- Segunda etapa, implantación en un escenario real.....</b>	<b>72</b>
7.2.1.- Desarrollos adicionales a la implantación.....	72
<b>CAPÍTULO 8- CONCLUSIONES.....</b>	<b>75</b>
<b>CAPÍTULO 9- BIBLIOGRAFÍA Y REFERENCIAS.....</b>	<b>79</b>
<b>CAPÍTULO 10- ANEXOS.....</b>	<b>81</b>
10.1.- Manual de uso de waterLU.....	81
10.2.- Manual de uso de nuevas funcionalidades de AguasCli.....	124

## Índice de tablas

Tabla 1: Especificaciones de Datalogic Blackjet.....	13
Tabla 2: Especificación IP contra partículas.....	13
Tabla 3: Especificación IP contra agua.....	13
Tabla 4: Especificaciones del lector de códigos de barras.....	59
Tabla 5: Especificaciones de impresora Zebra S4M ZPL 200 dpi.....	66

## Índice de ilustraciones

Ilustración 1: Estructura informática.....	3
Ilustración 2: Contadores con número de serie.....	7
Ilustración 3: Código de barras.....	7
Ilustración 4: Datalogic Blackjet.....	11
Ilustración 5: Contador.....	15
Ilustración 6: Lectura Actual.....	16
Ilustración 7: Ruta.....	16
Ilustración 8: Orden Ruta.....	17
Ilustración 9: Operario.....	17
Ilustración 10: Modelo de datos.....	18
Ilustración 11: Identificación de contador.....	19
Ilustración 12: Registro de lectura.....	19
Ilustración 13: Registro de incidencia.....	19
Ilustración 14: Seguimiento de ruta.....	19
Ilustración 15: Lectura de contador.....	20
Ilustración 16: Sincronización.....	20
Ilustración 17: Configurar.....	20
Ilustración 18: Teclado en pantalla.....	22
Ilustración 19: Navegación en waterLU.....	23
Ilustración 20: Botones F1 y F2.....	24
Ilustración 21: Pantalla identificación.....	24
Ilustración 22: Pantalla menú principal sin configuración.....	25
Ilustración 23: Pantalla menú principal con configuración.....	25
Ilustración 24: Pantalla de selección de ruta.....	25
Ilustración 25: Pantalla de selección de contador.....	26
Ilustración 26: Pantalla de toma de lecturas sin datos.....	26
Ilustración 27: Pantalla de toma de lecturas con datos.....	26
Ilustración 28: Ajustador con valores 100 y 20.....	27
Ilustración 29: Ajustador con valores 10 y 2.....	27
Ilustración 30: Aviso medida baja.....	27
Ilustración 31: Aviso medida alta.....	27
Ilustración 32: Aviso de relectura.....	28
Ilustración 33: Pantalla de registro de incidencias.....	28

Ilustración 34: Pantalla de configuración general.....	28
Ilustración 35: Pantalla de sincronización.....	29
Ilustración 36: Pantalla de configuración de sincronización.....	29
Ilustración 37: Pantalla de carga.....	30
Ilustración 38: Pantalla de abandono de la aplicación.....	30
Ilustración 39: Pantalla de cambio de contraseña.....	31
Ilustración 40: Pantalla de bloqueo.....	31
Ilustración 41: Gestores y su dependencia del programa.....	32
Ilustración 42: Confirmación de inicio de sincronización.....	42
Ilustración 43: Selección de la fecha de las lecturas e incidencias.....	42
Ilustración 44: Resumen de registro de lecturas e incidencias.....	43
Ilustración 45: Progreso, desconexión y tiempo invertido.....	43
Ilustración 46: Ejecución de servicios en OlivaNova.....	44
Ilustración 47: Perversión de la ejecución del servicio.....	45
Ilustración 48: Acceso a impresión de código de barras.....	45
Ilustración 49: Diálogo de selección y configuración de impresora.....	46
Ilustración 50: Código de barras obtenido.....	46
Ilustración 51: Movimiento de datos en sincronización.....	47
Ilustración 52: Fases de sincronización.....	47
Ilustración 53: Protocolo de sincronización en dispositivos.....	48
Ilustración 54: Espera para ejecución de operación.....	48
Ilustración 55: Protocolo de sincronización en AguasCli.....	49
Ilustración 56: Operación actualizar lecturas en dispositivo.....	50
Ilustración 57: Operación actualizar lecturas en AguasCli.....	50
Ilustración 58: Operación actualizar información en dispositivo.....	51
Ilustración 59: Operación actualizar información en AguasCli.....	51
Ilustración 60: Operación actualizar configuración en dispositivo.....	52
Ilustración 61: InformationGetter.....	55
Ilustración 62: Producción de datos.....	56
Ilustración 63: Error de sincronización en dispositivo.....	57
Ilustración 64: Botones para cancelar sincronización.....	57
Ilustración 65: Código de barras numérico.....	60
Ilustración 66: Código de barras alfanumérico.....	60
Ilustración 67: Códigos de barras lineales.....	61
Ilustración 68: Código de barras 2D.....	61
Ilustración 69: Formato Código 39.....	61
Ilustración 70: Formato Código 128.....	62
Ilustración 71: Formato Entrelazado 2 de 5.....	62
Ilustración 72: Formato UPC.....	62
Ilustración 73: Formato EAN13.....	62
Ilustración 74: Formato DataMatrix.....	63
Ilustración 75: Formato PDF147.....	63
Ilustración 76: Impresora Zebra S4M ZPL 200 dpi.....	65
Ilustración 77: Formato de etiqueta y código de barras.....	67





# CAPÍTULO 1- Introducción

Desde la aparición de las computadoras y los sistemas informáticos se han buscado y encontrado maneras de adaptar la informática a las empresas con el propósito de mejorar los procesos de negocio y los servicios. Los campos más frecuentes de aplicación de sistemas informatizados se encargan de realizar y monitorizar tareas mecánicas, y facilitar la ejecución de otras tareas.

Con la llegada de los dispositivos móviles y técnicas avanzadas de comunicación a estos sistemas informatizados las empresas encuentran nuevas aplicaciones y campos en los que emplear los sistemas informáticos.

Gracias a la potencia de cálculo y de almacenamiento de los dispositivos actuales junto con sus capacidades inalámbricas de comunicación y la variedad de accesorios disponibles son pocos los campos o áreas en las que la aplicación de dispositivos móviles no reporte un beneficio claro.

En este proyecto se aplicará el uso de dispositivos móviles en el campo de la lectura de contadores de agua, procurando modernizar y automatizar el proceso empleado hasta ahora de toma de lecturas. La introducción de los dispositivos en este campo afectará a otras partes del sistema, y se trabajará para conseguir un funcionamiento perfecto entre todas las partes, con el objetivo de mejorar en gran medida el desarrollo de las tareas en la lectura de contadores.

## 1.1.- Aguas del Bullent, S.A.

Este sistema ha sido desarrollado para la empresa **AGUAS DEL BULLENT, S.A.** (<http://www.aguasdelbullent.com>), con sede en **Oliva, Valencia, España**. Esta empresa se dedica al suministro de agua potable en el área de la playa de Oliva, ofreciendo servicio a cerca de 8.000 viviendas, a lo largo de 11 km de playa, dando servicio a más de 40.000 personas en temporada alta. Cuenta hoy en día con aproximadamente 3.000 contadores dados de alta.

La empresa cuenta con unas oficinas centrales donde se lleva a cabo la gestión, y con varios operarios encargados de las tareas de instalación, mantenimiento y lectura de contadores. Este proyecto pretende introducir en la empresa el uso de dispositivos móviles para asistir en la tarea de toma de lecturas de contadores.

## 1.2.- Descripción del documento

Este documento describe el proceso y trabajo realizado para aportar una solución a la necesidad de **Aguas del Bullent S.A.** a mejorar su proceso de toma de lecturas. Los apartados destacados son éstos:

- En el [capítulo 1](#) es presentado el trabajo de este proyecto según la estructura del documento.
- En el [capítulo 2, Contexto y tecnología](#), se presenta el sistema actual que emplea **Aguas del Bullent S.A.** generando mediante la metodología **OLIVANova**. El proceso de toma de lecturas que se pretende mejorar es expuesto, para pasar a enumerar las posibles soluciones que se barajaron y explicitar cuáles de ellas finalmente fueron adoptadas; en concreto, la

introducción de dispositivos móviles para el trabajo de campo, la ampliación de la funcionalidad del sistema existente y la identificación de contadores mediante códigos de barras.

- En el [capítulo 3](#), **Dispositivos móviles**, se introduce la plataforma escogida teniendo en cuenta la solución adoptada, y se enumeran las limitaciones que impone y las ventajas que reporta. Además se especifican los requerimientos funcionales de esta parte del sistema y el diseño escogido para satisfacerlos. Finalmente se muestra la aplicación resultante tras el desarrollo haciendo hincapié en la adaptación de la interfaz al perfil del usuario final.
- En el [capítulo 4](#), Aplicación **OLIVANova**, se describe el sistema actualmente en funcionamiento, [AguasCli](#), y su carácter evolucionable. Se especifica la nueva funcionalidad requerida en el sistema para conformar la solución adoptada, al tiempo que se presentan los mecanismos empleados con el fin de mantener al mínimo los cambios realizados sobre el código original. Finalmente se muestran el resultado de la implementación de las nuevas funcionalidades.
- En el [capítulo 5](#), Sincronización, se describe el proceso crítico de sincronización, especificando el protocolo diseñado y enumerando las operaciones de intercambio de datos que forman parte del proceso. Además se define el papel que juega la aplicación de escritorio [AguasCli](#) en la comunicación entre el dispositivo y el servidor central. Se explican las técnicas empleadas para controlar la carga del servidor central y el comportamiento adoptado ante errores y cancelaciones.
- En el [capítulo 6](#), Códigos de barras, se exponen los diferentes tipos de códigos de barras existentes y los motivos de haber seleccionado EAN13 como el apropiado para este sistema. Se realiza una descripción de las etiquetas a emplear considerando las condiciones de uso en lugares húmedos y a la intemperie. Por último se comenta la funcionalidad que el uso de códigos de barras aporta en el dispositivo y requiere de la aplicación [AguasCli](#).
- En el [capítulo 7](#), Implantación, se relata el proceso de implantación del sistema, realizado en dos etapas. Una primera etapa en la que se recogen cambios y sugerencias con tal de conseguir un funcionamiento que se ajuste a la forma de trabajo llevada hasta ahora y una fácil transición; y una segunda etapa de integración del sistema en la infraestructura existente, reemplazando antiguos métodos y añadiendo nuevas funciones.
- En el [capítulo 8](#), Conclusiones, se resume el trabajo realizado destacando los resultados obtenidos.
- En el [capítulo 9](#), Bibliografía, se detallan las referencias empleadas en el documento.
- En los [anexos](#) se incluyen los documentos y manuales desarrollados.



## CAPÍTULO 2- Contexto y tecnología

En este capítulo se describe el entorno en el que este proyecto cobra forma y fue desarrollado, comenzando con la metodología **OLIVANova** [1][2] empleada para construir el sistema actualmente en funcionamiento en **Aguas del Bullent S.A.** A continuación se mostrará la estructura e infraestructura de funcionamiento del sistema tal y como está realmente implantado, debido en parte a la filosofía de la metodología empleada.

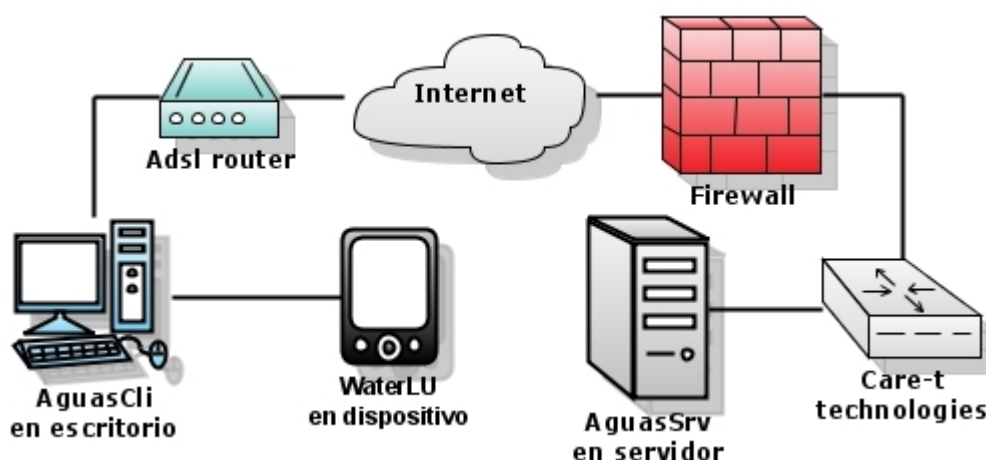
A seguir se expone en detalle el proceso de toma de lecturas de contadores que pretende mejorarse, destacando los puntos que pueden mejorarse notablemente gracias a la aplicación de un sistema informático; pasando a enumerar posibles soluciones que permiten llevar a cabo la misma tarea de una manera más eficiente. Cada una de estas soluciones conlleva unas ventajas diferentes, así como unos costes y una complejidad también diferentes.

Por descontado, una única solución será la que se adoptará; y ésta habrá sido elegida entre las propuestas por reportar beneficios y mejoras al proceso manteniendo un coste y complejidad aceptables.

### 2.1.- Estructura informática existente

La estructura informática empleada en **Aguas del Bullent S.A.** es en parte resultado del sistema obtenido mediante **OLIVANova**. Un modelo **OLIVANova** da como resultado un sistema cliente servidor cuando es transformado de modelo a aplicación funcional. De esta forma se tienen dos partes del sistema que pueden ejecutarse independientemente, y que permite el acceso de múltiples clientes a un mismo conjunto de datos y servicios que controla un único servidor.

Disponiendo de un sistema dividido en cliente y servidor podemos montar la estructura como mejor convenga a nuestras necesidades. Desde instalar ambas partes en un único equipo, de forma que todo el sistema funciona localmente en un equipo, a instalar en servidor en un equipo y uno o más clientes en varios equipos, de forma que tenemos un sistema distribuido con acceso concurrente a un conjunto de datos y servicios.



*Ilustración 1: Estructura informática*

En **Aguas del Bullent S.A.** han optado por mantener una configuración distribuida en la que el servidor se encuentra situado en otra empresa y mantenido por ellos, protegido en su red por sistemas de cortafuegos. Así en las oficinas de **Aguas del Bullent S.A.** dispone de un equipo con el cliente instalado de forma que pueden acceder a los datos y servicios del servidor. Por ahora disponen de un único cliente, pero sin mayores problemas podrían incluir más si fuera necesario.

El entorno que encontramos en **Aguas del Bullent S.A.** hace poco recomendable modificar la estructura de la parte servidora, o intentar añadir funcionalidad extra en la parte servidora. Esta estructura afectará a la elección de una u otra solución.

### **2.1.1.- Cliente servidor**

El sistema que se obtiene de un modelo **OLIVANova** sigue una estructura clásica de cliente servidor.

El servidor se encarga del nivel de datos empleando un gestor de bases de datos, así como del nivel de lógica de negocio ofreciendo los servicios definidos en el modelo que pueden ser invocados remotamente para introducir, modificar, manipular o borrar datos del sistema. De las diferentes opciones de transformación para el servidor de **Aguas del Bullent S.A.** se optó por la transformación del modelo a código **C#**, de forma que el servidor está funcionando sobre la plataforma **.Net framework**.

El cliente se hace cargo del nivel de interfaz, y cuenta con un sistema de identificación y roles que limitan el acceso a solo el personal autorizado, mostrando una vista con controles y datos diferente para cada rol. De la transformación de la parte cliente puede resultar una interfaz en una aplicación en uno de varios lenguajes, o bien en una interfaz para acceso vis Web. Para el sistema de **Aguas del Bullent S.A.** se optó por una transformación a una aplicación **C#**, de forma que el cliente está funcionando sobre la plataforma **.Net framework**.

## **2.2.- Proceso de toma de lecturas**

El objetivo principal de este proyecto es mejorar el proceso de toma de lectura de consumo de contadores. Para ello, es obvio, es necesario conocer como se realiza el proceso a mejorar. Intentaremos encontrar los puntos donde la aplicación de tecnologías informáticas reporte la mayor mejoría.

Antes de la implantación de este nuevo sistema, el proceso de toma de lecturas estaba caracterizado por :

- **Lecturas cada trimestre** : cada trimestre se lleva a cabo una toma de lecturas en todos los contadores dados de alta en la empresa.
- **Recorrido por rutas** : la toma de lecturas se realiza según un orden establecido en las rutas definidas y en el orden de los contadores en cada ruta. Proporciona organización al proceso así como los caminos óptimos a seguir durante las sesiones de medición.
- **Soporte** : se emplea un libro de hojas en las que se ha impreso los datos de los contadores y donde se recogerán los datos de la toma de lecturas. Este libro, con los aproximadamente 3000 contadores dados de alta llega a tener un volumen poco manejable, y el problema se agravará con el crecimiento de los contadores adscritos a la empresa.

- **Búsqueda manual** : durante las sesiones de toma de lecturas, cada contador ha de ser buscado manualmente y sin ningún tipo de ayuda en el libro que contiene los datos de los contadores. Gracias a la organización en rutas esto no suele ser un problema hasta que se desea tomar lectura de un contador que no se encuentra en la ruta seguida.
- **Escritura manual de lecturas** : los datos recogidos de la toma de lecturas son apuntados manualmente en el libro lo que da lugar a errores.
- **Escritura manual de incidencias** : las incidencias detectadas durante las sesiones de toma de lecturas son registradas manualmente en el libro, pero no llegan a registrarse en el sistema informatizado puesto que no da soporte para ello.

El tratamiento manual de los datos combinado con el gran volumen de ellos que se maneja da lugar a :

- 1) que el proceso cueste mucho tiempo de terminar, llevando entre 4 y 5 semanas para tomar la lectura de todos los contadores, y
- 2) que se produzcan errores en la toma de lecturas, ya sea registrando un valor de lectura incorrecto como registrándolo en el contador equivocado.

Conociendo el proceso podemos comenzar a buscar posibles mejoras o diferentes acercamientos al proceso de toma de lecturas. De las opciones que encontremos habremos de optar por una de ellas que será la que finalmente se implantará.

### 2.3.- *Propuestas*

Desde un principio existe el objetivo de automatizar al máximo el proceso de toma de lecturas, con la finalidad de ahorrar tiempo y reducir errores. Ahora bien, ese objetivo no puede alcanzarse a cualquier precio. Debemos tener en mente a todas horas que la solución a adoptar no puede ser más compleja o costosa que lo que la empresa **Aguas del Bullent S.A.** pueda o quiera aceptar.

Antes de considerar cualquier propuesta, ésta ha de dar solución a :

- **Identificación** : la identificación de un contador debe poder realizarse de la manera más automática posible y que en la medida de lo posible no requiera introducción de datos por parte del operario para evitar errores manuales.
- **Toma de lecturas** : la toma de lecturas debe poder realizarse de la manera más automatizada posible, evitando en lo posible la introducción manual del valor.
- **Registro incidencias** : debe proporcionarse soporte al registro y trato de la incidencias ocurridas en las sesiones de toma de lecturas.
- **Integración** : la integración con el sistema ya existente debe ser lo menos compleja posible, y en la medida de lo posible no debe modificar el sistema ya implantado.

Con estas consideraciones fijadas las opciones que se barajaron fueron :

- **Sistemas de telelectura de larga distancia** : [3]

Estos sistemas permiten la lectura de un contador sin intervención humana y a larga distancia. Consta de un módulo por contador que lee el valor de la lectura tal y como aparece en el contador y lo transmite a larga distancia mediante radiofrecuencia, GSM o PSTM. Tanto la identificación como la

toma de la lectura se realizan automáticamente, desde largas distancias, sin intervención de los operarios y a cualquier hora.

Ahora bien, el registro de incidencias puede suponer un problema en este sistema puesto que los módulos de telelectura no pueden detectar todas las incidencias posible, requiriendo la intervención de un operario para ese caso.

Además, el coste y complejidad que supondría integrar este sistema en el ya existente es demasiado elevado como para que resulte aceptable. Esto hace que esta propuesta no sea viable para mejorar el proceso de toma de lecturas.

- **Sistemas de telelectura de corta distancia : [4]**

Estos sistemas funcionan de forma similar a los de telelectura de larga distancia sólo que limitados en la distancia a la que pueden transmitir los datos. La distancia de emisión suele ser suficiente como para poder realizar la toma de lecturas desde un vehículo sin necesidad de abandonarlo ni de visitar el contador.

Las ventajas que nos aporta este sistema son similares a las que nos aportan los sistemas de telelectura de larga distancia, y también son similares los problemas que plantea.

En este sistema el coste y complejidad de la integración con el sistema ya existente siguen estando fuera del rango aceptable. Esta propuesta tampoco es viable como mejora al proceso de toma de lecturas.

- **Dispositivos móviles : [5]**

Este sistema propone emplear dispositivos móviles como nuevo soporte en la toma de lecturas. La intervención del operario sigue siendo necesaria, no obstante podemos eliminar o reducir la necesidad de introducir datos manualmente. La identificación de los contadores puede automatizarse empleando el seguimiento de rutas y/o métodos de escaneo que identifiquen inequívocamente a los contadores; mientras que la toma de lecturas puede proporcionar ayuda en la introducción y validación de datos o adquirirlos directamente con técnicas de escaneo.

La intervención del operario nos da apoyo a un buen soporte para el registro de incidencias, ya que el operario es capaz de detectar cualquier tipo de incidencias que puede ocurrir.

El coste de este sistema es una pequeña parte de los anteriores propuestos y aceptable para la empresa. La complejidad de integración no supone más que una manipulación y sincronización de datos entre los dispositivos y el servidor central.

Esta propuesta aplica mejoras informáticas al proceso de toma de lecturas allí donde el proceso más las necesita al tiempo que mantiene un coste y complejidad aceptables para la empresa.

De las propuestas planteadas la que puede ofrecernos una mejora dentro del rango de coste y complejidad aceptable por la empresa es la introducción de dispositivos móviles que, en conjunto con una aplicación desarrollada expreso para este proceso, pueden ayudar a reducir el tiempo y los errores en el proceso de toma de lecturas.

A continuación se describe más en detalle como pueden aplicarse dispositivos móviles al proceso, con las tecnologías a emplear, las nuevas funcionalidades a implantar y los cambios necesarios en el sistema.

### 2.3.1.- Dispositivos móviles

El empleo de dispositivos móviles nos permite trasladar parte del sistema directamente al campo de trabajo. Sustituyendo el anterior soporte de papel por un dispositivo móvil ganamos en comodidad, lo que se traduce en un menor tiempo para llevar a cabo la tarea. De la misma manera utilizando diferentes añadidos o accesorios del dispositivo móvil podemos automatizar tareas con el fin de eliminar errores.

Una vez decidido la utilización de dispositivos móviles para el apoyo en la toma de lecturas el siguiente paso consiste en establecer las automatizaciones a incorporar y como incorporarlas.

Para automatizar la identificación de contadores existen diferentes opciones, cada cual con diferentes características :

- **Identificación por OCR** : mediante una cámara adaptada en el dispositivo [6] se identificará el contador extrayendo con técnicas de OCR el número de serie del contador a partir de una imagen tomada de éste.



*Ilustración 2: Contadores con número de serie*

Podemos ver que la situación del número de serie que identifica al contador varía de un modelos a otro. Ésto sumado a los diferentes colores, la precisión de las cámaras disponibles actualmente, la inclinación variable y la baja luminosidad presente en el entorno de trabajo hace poco viable el reconocimiento OCR para la identificación de un contador.

- **Identificación por código de barras** : cada contador dispondrá de un código de barras con un número que permite su identificación leyendo el código de barras mediante un lector adaptado al dispositivo [7].



*Ilustración 3: Código de barras*

La tecnología de lectura de códigos de barras está actualmente ampliamente desarrollada, y es fácil encontrar dispositivos que incorporen un lector de códigos de barras. Los lectores actuales pueden leer un códigos de barras en condiciones que no sean óptimas con cierta facilidad, incluso aquellos ya deteriorados por humedad o desgaste en la intemperie.

La producción de códigos de barras que soporten las condiciones del entorno de trabajo es sencilla y su coste no es elevado, por lo que la identificación con códigos de barras se presenta como una opción viable.

- **Identificación por ruta** : realizando un seguimiento de ruta puede identificarse un contador por el orden que tienen los contadores en las rutas.

La organización ya existente de los contadores en diferentes rutas nos proporciona una base para ofrecer una identificación de contadores. Partiendo de la selección de una ruta a seguir podemos identificar cada contador en la ruta según van midiéndose siguiendo el orden de contadores. Esta opción puede existir junto con otra opción de identificación automatizada.

- **Identificación manual** : no es una alternativa de identificación automática, aún así esta alternativa puede ofrecerse en conjunto con otra de identificación automática.

Para automatizar la toma de lecturas existen varias alternativas, cada una de ellas con diferentes características :

- **Lectura por OCR** : mediante una cámara adaptada en el dispositivo puede tomarse una imagen del contador de la que se extraerá mediante técnicas de OCR el valor de la lectura.

Esta alternativa además de presentar los mismos problemas que la identificación del contador por OCR suma la posibilidad de encontrar un dígito en transición al siguiente, con lo que el reconocimiento del valor correcto se dificulta aún más.

El reconocimiento por OCR no se presenta como una opción viable.

- **Estimación estadística del consumo** : realizando cálculos sobre el histórico de consumo de cada contador podemos establecer previsiones de consumo esperado así como rangos de valores para una lectura correcta. Esta alternativa no elimina la interacción del operario, y será viable siempre que pueda conseguirse un sistema de cálculo de previsión adecuado así como facilidades en el ajuste del valor calculado.
- **Introducción manual** : no es una forma automática de introducir el valor en el sistema, aún así esta alternativa puede ofrecerse en conjunto con una introducción del valor de la lectura automática.

Para el registro de incidencias los dispositivos pueden proporcionar un sistema de registro de incidencias. El operario al detectar una incidencia registraría en el dispositivo el tipo de incidencia ocurrido, y posteriormente esa información se trasladaría al servidor central.

La integración de los dispositivos en el sistema existente no plantea un problema mayor que el traslado de los datos de los dispositivos al servidor central. La aplicación que desarrollemos para los dispositivos deberá ofrecer una solución al traslado de los datos desde el dispositivo al servidor central y viceversa.

### 2.3.2.- Sistema actual

Habiendo adoptado como solución para campo el uso de dispositivos móviles que asistirán en la tarea de toma de lecturas se hace necesario revisar el sistema existente para encontrar los cambios o adiciones que habrá que hacer.

Como se ha comentado, los cambios o adiciones en la parte servidora se evitarán siempre que sea posible, y los cambios manuales en el código se reducirán

al mínimo, procurando introducir los cambios en el modelo **OLIVANova**.

Con tal de mejorar el proceso de toma de lecturas el sistema ya existente debe dar soporte a :

- **Cálculo de previsiones** : la información de un contador ha de incluir el cálculo de la previsión de consumo tal y como la necesita el dispositivo. **OLIVANova** permite establecer atributos que se encarguen de calcular una estimación del consumo basándose en el histórico. Es un cambio que aplicaremos directamente al modelo.
- **Registro de incidencias** : el sistema ha de soportar el registro y tratamiento de incidencias. Actualmente el sistema no ofrece soporte al registro de incidencias. En el modelo **OLIVANova** puede definirse esta nueva funcionalidad sin necesidad de cambios manuales.
- **Impresión de códigos de barras** : unas de las opciones de identificación de contadores implica el uso de códigos de barras. Ha de ofrecerse algún método de obtener un código de barras a partir de un contador. El modelo **OLIVANova** no permite definir este tipo de operación, por lo que aplicaremos manualmente los cambios que sean necesarios con tal de ofrecer funcionalidad para imprimir códigos de barras a partir de contadores.
- **Configuración de dispositivos** : la aplicación de los dispositivos tendrá varios parámetros de configuración, y han de ser configurables desde la parte cliente del sistema, de forma que pueda compartirse una misma configuración para los distintos dispositivos de que se disponga. En el modelo podemos especificar nuevas clases que se encarguen de esta funcionalidad.

Como puede deducirse, el cálculo de previsiones, el registro de incidencias y la configuración de dispositivos son funcionalidades nuevas que serán añadidas al modelo, ya que el modelado en **OLIVANova** tiene la capacidad de hacerlo. Ahora bien, la impresión de códigos de barras es una funcionalidad que habrá que implementar manualmente y después añadir a la aplicación, procurando interferir poco en el código de la aplicación.

### 2.3.3.- Sincronización

El dispositivo necesita información del servidor central para llevar a cabo las tareas de toma de lecturas; y a su vez recogerá información cuyo destino final es el servidor central, donde será procesada y tratada.

Se impone la necesidad de diseñar e implementar un algoritmo de sincronización que permita el flujo de la información entre dispositivos y servidor central. El modelado en **OLIVANova** no puede proporcionarnos una solución, por lo que diseñaremos e implementaremos una solución para permitir la sincronización de datos.





## CAPÍTULO 3- Dispositivos móviles

En este capítulo se describe la parte de diseño y desarrollo del sistema que involucra a los dispositivos móviles, o PDAs. Se describe en detalle el dispositivo escogido, el **Datalogic Blackjet™** [5]; así como el diseño de la aplicación **waterLU**, desde los niveles que componen la aplicación al resultado final pasando por la adaptación que se ha realizado para que el sistema resulte fácil e intuitivo de emplear para el usuario destino.

Los dispositivos móviles y el sistema diseñado y desarrollado para ellos juegan un papel importante en conseguir que el proceso de toma de lecturas de contadores mejore. El diseño y desarrollo del sistema se ha centrado en emplear en la medida de lo posible las características más avanzadas de **Datalogic Blackjet™**, pero también preparando el sistema para su uso en dispositivos más genéricos.

### 3.1.- Dispositivos

La elección de los dispositivos móviles a emplear debía tomar en consideración varios factores. Las condiciones de trabajo y las necesidades del sistema a desarrollar imponen unas restricciones a las que los dispositivos han de amoldarse.

Por el entorno de trabajo y la elección de emplear códigos de barras los dispositivos móviles convencionales no son los más apropiados, recomendando el uso de dispositivos móviles industriales que garanticen su adaptación a las condiciones de trabajo de esta tarea.

Fue la empresa la encargada de comprar los dispositivos más adecuados a sus necesidades que cumplieran con los requisitos del sistema/solución. Finalmente los dispositivos para los que se diseñaría y desarrollaría el sistema/aplicación son tres dispositivos **Datalogic Blackjet™**.



*Ilustración 4: Datalogic Blackjet*

#### 3.1.1.- Datalogic Blackjet

Los dispositivos **Datalogic Blackjet™** tienen estas especificaciones técnicas :

<b>Especificación eléctrica</b>	
Alimentación	
Alimentación DC	12 a 14 V $\pm$ 5%
Batería	Li-Ion 1070 mAh@ 7.4 V de 2 celdas
Batería gran capacidad	Li-Ion 1800 mAh@ 7.4 V de 2 celdas
Batería interna reserva	Li-Ion 30 mAh recargable
<b>Especificación de comunicación</b>	
Interfaz Bluetooth	--
Interfaz serie	RS232 USB 1.1
<b>Especificación de comunicación inalámbrica</b>	
WLAN	IEEE 802.11b DSSS
WPAN	Bluetooth ® IEEE 802.15, Class 2, Version 1.2
WWAN	GSM/GPRS Tri-band (900, 1800, 1900 mHz) Radio con interfaz de tarjeta SIM
<b>Especificación ambiental</b>	
Temperatura de funcionamiento	-10° a +50°C
Temperatura de almacenamiento	-20° a 70°C
Humedad	90% sin condensación a temp. < 40°C
Protección	IP64
Protección ESD	4 KV descarga al contacto, 8 KV descarga aérea
Protección ante caídas	1,5 metros
<b>Especificación hardware</b>	
FLASH	128 MB
DRAM	128 MB
Microprocesador	Intel XScale PXA 270 520 MHz
Espacio tarjetas	Mini SD
Audio	Altavoz, micrófono, Jack para auriculares
LED	LED programable de dos colores Indicador de carga (dos colores) Bluetooth® (LED azul)
Pantalla	TFT LCD, resolución 240x320 píxeles de 64K colores con retroiluminación
Teclado	25 teclas de plástico iluminadas + botón de encendido + 2 botones laterales

<b>Especificación mecánica</b>	
Dimensiones (LARGOxANCHOxALTO)	17,6x9x2,8 cm
Peso	424 g
<b>Especificaciones de programación</b>	
Sistema operativo	Windows CE .Net 5.0

*Tabla 1: Especificaciones de **Datalogic Blackjet***

En las características técnicas puede observarse como los dispositivos son adecuados para las condiciones que se encuentra en las áreas de trabajo donde se llevan a cabo las lecturas de contadores.

Una factor importante para utilizar este tipo de dispositivos es el factor de protección **IP64** que presenta. La protección **IP** (Ingress Protection [8]) es un estándar internacional de la organización **IEC** (International Electrotechnical Commission [9]) que define el grado de protección que un aparato eléctrico ofrece ante la intrusión de objetos sólidos, contacto accidental, agua o polvo.

Los grados de protección se definen mediante dos dígitos, el primero de ellos indicando la protección ante partículas físicas y el segundo la protección ante agua. Estas tablas definen la relación entre el dígito y la protección ofrecida:

<b>Protección contra partículas</b>		
<b>Nivel</b>	<b>Tamaño</b>	<b>Efectivo contra</b>
0	--	Ninguna protección.
1	> 50 mm	Partes grandes del cuerpo.
2	> 12,5 mm	Dedos o similar.
3	> 2,5 mm	Herramientas, cables gruesos.
4	> 1 mm	La mayoría de cables y tornillos.
5	Protección contra polvo	El polvo no interferirá en el equipo.
6	Hermético	El polvo no entrará.

*Tabla 2: Especificación **IP** contra partículas*

<b>Protección contra agua</b>		
<b>Nivel</b>	<b>Protegido contra</b>	<b>Detalles</b>
0	Sin protección	--
1	Gotas	Gotas cayendo verticalmente.
2	Gotas girado 15º	Gotas cayendo verticalmente con el dispositivo a 15º.
3	Aguas rociada	Aguas cayendo hasta a 60º.
4	Agua salpicada	Aguas cayendo en cualquier dirección.
5	Chorro de agua	Agua proyectada contra el dispositivo.
6	Agua pesada	Agua proyectada en chorro a presión.
7	Inmersión 1 m.	Inmersión del dispositivo hasta 1m
8	Inmersión + 1 m	Inmersión indicada por el fabricante.

*Tabla 3: Especificación **IP** contra agua*

En las tablas se comprueba que los dispositivos **Datalogic Blackjet™** están protegidos contra polvo y agua en grados que permiten su utilización en el entorno de trabajo. Además cuentan con protección contra caídas, que siempre son inevitables, y con un rango de funcionamiento de temperaturas más que satisfactorio. En resumen, estos dispositivos están preparados para funcionar sin preocupaciones en el entorno de trabajo que nos incumbe.

### 3.1.2.- Limitaciones y ventajas del dispositivo

Al utilizar los dispositivos móviles **Datalogic Blackjet™** podemos llevar el sistema informático hasta el mismo lugar en donde se realiza la toma de lecturas de los contadores. Sin embargo, estos dispositivos difieren bastante de los equipos informáticos estándar de escritorio. Estas diferencias imponen ciertas limitaciones que tendremos que tener en cuenta a la hora de diseñar y desarrollar el sistema; pero también aportan ciertas ventajas que no tendríamos con un equipo de sobremesa.

En cuanto a las limitaciones, cabe destacar éstas:

- **CPU** : el dispositivo cuenta con un microprocesador **Intel XScale PXA 270 520 Mhz**; procesador más que aceptable para realizar tareas comunes de interacción con el usuario, que sin embargo en tareas pesadas de cálculo muestra sus deficiencias. Por tanto evitaremos al máximo cargar el sistema del dispositivo con operaciones pesadas de mucho cálculo, limitándolo a un sistema de interacción y almacenaje.
- **Memoria** : el dispositivo cuenta con **128 MB** de memoria **DRAM**, cantidad suficiente para tener en ejecución el sistema operativo **Windows CE .Net 5.0**, el servidor de bases de datos **SQL Server Everywhere** y la aplicación desarrollada. No obstante, es recomendable no mantener en ejecución otros servicios o aplicaciones debido a que la demanda de memoria del sistema puede crecer, y al no disponer el dispositivo de un sistema de memoria virtual, impedir la correcta ejecución del sistema diseñado.
- **Capacidad almacenamiento** : el dispositivo cuenta con **128 MB** de memoria **FLASH**. Tal cantidad nos permite almacenar tanto el sistema operativo como los servicios y la aplicación holgadamente. Sin embargo, la base de datos necesaria sería excesivamente voluminosa si la portáramos directamente de la empleada en el servidor central; por tanto crearemos una propia con únicamente los datos que sí necesitamos. Esto nos obligará a realizar transformaciones de los datos en la fase de sincronización.
- **Tamaño pantalla** : el dispositivo cuenta con una pantalla **TFT LCD de 240x320 píxeles** de resolución a **64K colores** con **retroiluminación**. Es un espacio muy limitado si lo comparamos con una pantalla estándar de sobremesa. Una interfaz diseñada para este dispositivo deberá ser concisa, mostrando la información de manera clara y sin redundancias. Especialmente este sistema debe prestar una mayor importancia en este punto para no alienar a los usuarios finales.
- **Batería** : el dispositivo cuenta con una batería **Li-Ion 1070 mAh@ 7.4 V de 2 celdas**, que proporcionan una duración aproximadamente 4 horas de uso continuado. Será prioritario intentar que el sistema consuma lo mínimo posible.

Las ventajas que podemos destacar son:

- **Movilidad** : principal ventaja de un dispositivo móvil. Permite llevar el sistema al propio lugar de trabajo.
- **Lector de códigos de barras** : el dispositivo incorpora un lector de códigos de barras muy útil para identificar rápidamente y sin errores los contadores.
- **Evita “equipaje”** : el dispositivo incorpora todo lo necesario para realizar las tomas de lecturas, evitando el “equipaje” que imponía el método anterior.
- **Automatización** : tareas antes repetitivas y mecánicas son realizadas ahora automáticamente y de forma transparente en el sistema.

En el diseño y desarrollo del sistema procuraremos adaptarnos a las limitaciones y sacar el máximo provecho de las ventajas del dispositivo **Datalogic Blackjet™** escogido.

### 3.2.- Diseño del sistema

El sistema a desarrollar emplea de almacenaje de datos, interacción del usuario y reproducción de la funcionalidad requerida. Fácilmente puede realizarse una traducción a un sistema clásico de tres niveles con base de datos, lógica de negocio e interfaz de usuario.

#### 3.2.1.- Nivel de datos

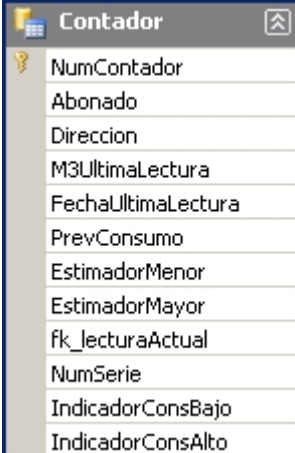
Los datos que emplea el sistema/aplicación son un subconjunto de los existentes en el servidor central, transformados con el fin de ocupar el mínimo espacio de almacenamiento en el dispositivo.

El modelo de datos para el sistema debe permitirnos almacenar información sobre los contadores de los que tomar lecturas, las lecturas que se toman, las rutas y la posición de los contadores en ellas y los operarios que utilizarán el sistema.

##### 3.2.1.a Modelo de datos

Intentando minimizar el espacio a ocupar manteniendo toda la información necesaria se decide utilizar estas clases/tablas que se relacionarán.

- **Contador** : Representa un contador, y contiene información para permitir su identificación, localización y estadísticas de consumo que asisten en la introducción de la lectura. Cuenta con estos campos:
  - *NumContador* (integer) : clave primaria, es el código de contador tal y como se encuentra en el servidor central.
  - *Abonado* (string) : cadena con el nombre y apellidos del abonado a quien está adscrito el contador. Derivado de la base de datos central.
  - *Dirección* (string) : cadena con la dirección y población del contador. Derivado de la base de datos central.



Contador	
NumContador	
Abonado	
Direccion	
M3UltimaLectura	
FechaUltimaLectura	
PrevConsumo	
EstimadorMenor	
EstimadorMayor	
fk_lecturaActual	
NumSerie	
IndicadorConsBajo	
IndicadorConsAlto	

*Ilustración 5:  
Contador*

- *M3UltimaLectura* (integer) : metros cúbicos que se leyeron del contador en la última lectura realizada. Derivado de la base de datos central. Es utilizado para validación de datos.
- *FechaUltimaLectura* (datetime) : fecha y hora en que se realizó la última lectura. Derivado de la base de datos central. Utilizado para cálculo de previsiones de consumo.
- *PrevConsumo* (integer) : previsión estimada del consumo realizada mediante análisis del histórico de consumo de este contador. Derivado de la base de datos central.
- *EstimadorMenor* (integer) : valor de la granularidad pequeña al hacer ajustes rápidos mediante botón o cursos.
- *EstimadorMayor* (integer) : valor de la granularidad grande al hacer ajustes rápidos mediante botón o cursos.
- *fk\_lecturaActual* (integer) : clave ajena a la tabla de **LecturaActual** indicando la lectura que se ha tomado para este contador. Utilizado a la hora de sincronizar las lecturas con el servidor central.
- *NumSerie* (string) : cadena que contiene el número de serie único del contador, tal y como se encuentra en la base de datos del servidor. Es un identificador alternativo. Se muestra junto al código de contador para facilitar la identificación.
- *IndicadorConsBajo* (integer) : estimador del mínimo consumo no anómalo obtenido de análisis del histórico de consumo de este contador. Derivado de la base de datos central.
- *IndicadorConsAlto* (integer) : estimador del máximo consumo no anómalo obtenido del análisis del histórico de consumo de este contador. Derivado de la base de datos central.

- **LecturaActual** : Representa una lectura tomada de un contador. Guarda información de consumo del contador, fecha y hora y posibles incidencias. Cuenta con estos campos :



LecturaActual	
id_lecturaActual	
M3Lectura	
FechaLectura	
CodigoIncidencia	
TextoIncidencia	

*Ilustración 6:  
Lectura Actual*

- *id\_lecturaActual* (integer) : clave primaria autoincrementada de la tabla. Referenciada por **Contador**.
- *M3Lectura* (integer) : consumo total, en metros cúbicos, leído del contador. Cantidad que será facturada por el servidor central. Este campo carece de significado si ha ocurrido una incidencia.
- *FechaLectura* (datetime) : fecha y hora de la toma de la lectura o incidencia.
- *CodigoIncidencia* (integer) : código numérico de la incidencia registrada al tomar la lectura de un contador. Este código indica el mensaje a registrar.
- *TextoIncidencia* (string) : cadena con el mensaje detallando el tipo de incidencia. Se registrará en el servidor central para ser tratado.

- **Ruta** : Representa una ruta, que agrupa contadores en un orden en el que deben ser leídos. Cuenta con estos campos :



Ruta	
CodRuta	
Nombre	

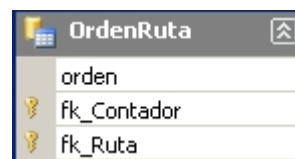
*Ilustración 7: Ruta*

- *CodRuta* (integer) : clave primaria de la tabla tal

y como aparece en el servidor central. Referenciado por **OrdenRuta**.

- *Nombre* (string) : cadena con el nombre que se mostrará de la ruta tal y como aparece en el servidor central.

- **OrdenRuta** : Representa la posición de un contador en una determinada ruta. Un contador podrá estar en más de una ruta y deberá estar en al menos una de ellas. Cuenta con estos campos :



OrdenRuta	
	orden
	fk_Contador
	fk_Ruta

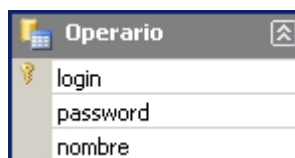
*Ilustración 8:  
Orden Ruta*

- *orden* (integer) : posición del contador *fk\_Contador* en la ruta *fk\_Ruta*, tal y como aparece en el servidor central.

- *fk\_Contador* (integer) : forma parte de la clave primaria de la tabla y es clave ajena al contador con posición *orden* en la ruta *fk\_Ruta*, tal y como aparece en el servidor central.

- *fk\_Ruta* (integer) : forma parte de la clave primaria de la tabla y es clave ajena a la ruta en la que el contador *fk\_Contador* se encuentra con posición *orden*, tal y como aparece en el servidor central.

- **Operario** : Representa un operario que puede identificarse en el sistema y realizar tareas de toma de lecturas y de sincronización. Cuenta con estos campos :



Operario	
	login
	password
	nombre

*Ilustración 9:  
Operario*

- *login* (string) : clave primaria, cadena con el texto de identificación del usuario ante el sistema, tal y como aparece en el servidor central.

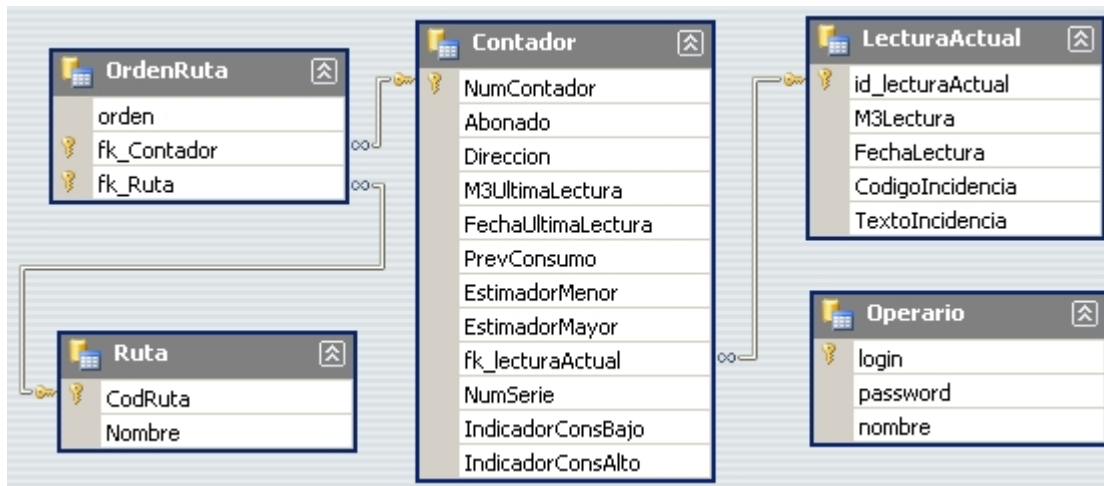
- *password* (string) : cadena codificada con la contraseña de autenticación del usuario ante el sistema, tal y como aparece en el servidor central.

- *nombre* (string) : cadena con el nombre del operario que le identifica en algunos mensajes o pantallas, tal y como aparece en el servidor central.

Estas tablas presentan relaciones que mantienen la estructura del modelo de datos. Así las relaciones existentes en el modelo de datos son estas:

- En **Contador** el campo *fk\_lecturaActual* es una clave ajena a la tabla **LecturaActual** referenciando cuál es la lectura tomada para ese contador. Un contador puede tener una única lectura.
- En **OrdenRuta** el campo *fk\_Contador* es una clave ajena a la tabla **Contador** referenciando de que contador es la posición que se indica para una ruta.
- En **OrdenRuta** el campo *fk\_Ruta* es una clave ajena a la tabla **Ruta** referenciando en que ruta se encuentra un contador y con qué posición.

El modelo de datos final, con todas las tablas y relaciones, que se empleará en el dispositivo será éste :



*Ilustración 10: Modelo de datos*

### 3.2.1.b Base de datos

El sistema manipulará un volumen de datos importante, más de 3.000 contadores (y creciendo) con sus correspondientes lecturas y posiciones en ruta conforman el grueso de la información con que trabajará el sistema. Tal cantidad de datos prácticamente nos obliga a emplear un sistema de bases de datos que se encargue de almacenar y gestionar eficazmente la información.

Se decidió emplear SQL Server 2005 Mobile [10] por la integración que presenta tanto con el entorno de desarrollo, con un conjunto de herramientas que facilitan el paso de diseño a desarrollo del modelo de datos, como con el sistema operativo **Windows CE .Net 5.0** de los dispositivos **Datalogic Blackjet™**.

### 3.2.2.- Lógica de negocio

El propósito básico del sistema es ofrecer la funcionalidad especificada en los requerimientos. El nivel de lógica de negocio se encargará de brindar esa funcionalidad. Este nivel se encuentra entre el nivel de datos y de interfaz, manipulando la información del nivel de datos con algoritmos y protocolos que implementan la funcionalidad, ofreciendo sus servicios al nivel de interfaz encargado de interaccionar con el usuario.

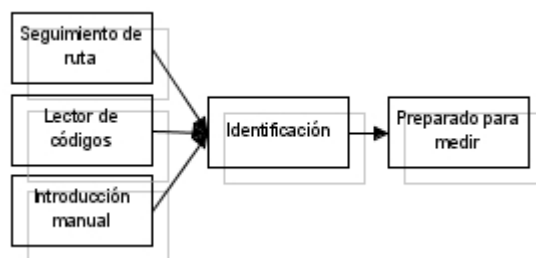
Las funcionalidades indispensables para este sistema se presentan aquí; cada una de ellas puede a su vez englobar varias funcionalidades que en conjunto conforman una funcionalidad mayor.

#### 3.2.2.a Toma de lecturas

Funcionalidad básica por la que se produce este sistema. La toma de lecturas está compuesta por varias funcionalidades que en conjunto permiten realizar la toma de una lectura de forma satisfactoria.

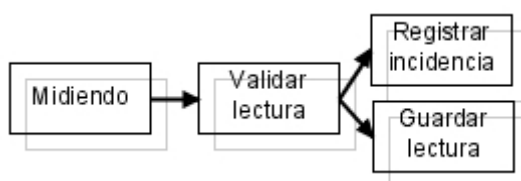
El primer paso para la toma de una lectura es siempre identificar el contador para el cuál se desea realizar esa lectura. Se ofrecen varios métodos para encontrar o identificar un contador, ya sea siguiendo una ruta, mediante lector de códigos de barras o con introducción manual del código.





*Ilustración 11: Identificación de contador*

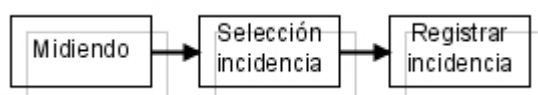
Con el contador ya identificado podemos guardar la lectura tomada, pero no sin antes validarla. La funcionalidad de validación está encapsulada y realizará, o bien el guardado de la lectura si es correcta, o bien registrará una incidencia en la lectura del contador.



*Ilustración 12: Registro de lectura*

### **3.2.2.b Registro de incidencias**

El registro de incidencias ocurridas durante la toma de lecturas es otra funcionalidad importante y secundaria a la de toma de lecturas. Este registro se realiza automáticamente hasta donde el dispositivo puede llegar, validando las lecturas y registrando las incidencias de valores anómalos. No obstante existen incidencias que debe detectar el operario. Para estas incidencias la funcionalidad de registro de incidencias tendrá que ser invocada explícitamente mediante el nivel de interfaz de usuario.

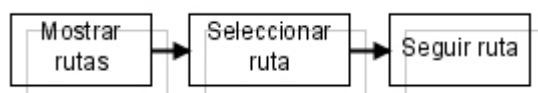


*Ilustración 13: Registro de incidencia*

### **3.2.2.c Seguimiento de rutas**

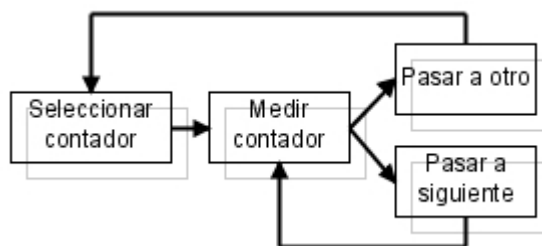
Funcionalidad que aporta comodidad y organización al sistema, permite planificar itinerarios y objetivos a cumplir por jornada. El seguimiento de rutas se presenta en dos etapas.

La primera etapa permite al usuario seleccionar la ruta a seguir de entre todas las posibles y, al seleccionar una de ellas, situarla como ruta a seguir.



*Ilustración 14: Seguimiento de ruta*

La segunda etapa, con una ruta ya seleccionada, consiste en seleccionar el contador a medir. De forma automática se sitúa como contador a medir el primero por leer de la ruta. De todas formas, facilitando la toma de lecturas, se permite efectuar una lectura de cualquier contador en cualquier momento, ya sea de un contador en la misma ruta saltando algunas posiciones o de un contador totalmente fuera de ruta, abandonando la ruta temporalmente.



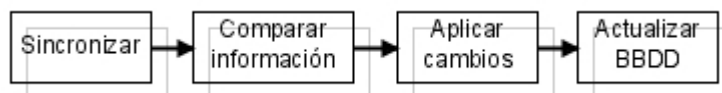
*Ilustración 15: Lectura de contador*

### 3.2.2.d Sincronización

Una funcionalidad crítica, es la encargada de mantener los datos del dispositivo actualizados respecto a los del servidor central y de proporcionar las lecturas tomadas para ser introducidas al servidor central. La sincronización consta de un grupo de funcionalidades que en conjunto consiguen mantener los datos actualizados.

La configuración de la sincronización afecta a las siguientes etapas de la sincronización, definiendo qué datos se actualizarán.

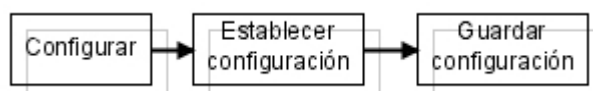
El proceso de sincronización comprende dos etapas secuenciales. Para la primera de ellas es necesario disponer de conexión. Durante esta primera etapa se realiza el intercambio de información entre el dispositivo y el servidor central, buscando cambios, nuevos datos y datos a eliminar. En la segunda etapa se aplican los cambios encontrados a la base de datos del dispositivo, actualizando realmente el sistema.



*Ilustración 16: Sincronización*

### 3.2.2.e Configuración

La configuración del sistema permite adaptar el comportamiento que el resto de funcionalidades y el nivel de interfaz tendrán ante determinadas acciones que pueden considerarse accidentales o anómalas. Así, para cada opción de la configuración puede establecerse el comportamiento como Allow (Permitir), Deny (Prohibir) o Prompt (Preguntar), instando al sistema a permitir, prohibir o preguntar frente a esas acciones inesperadas.



*Ilustración 17: Configurar*

El sistema espera encontrar la configuración definida en un archivo de texto claro. En caso de no encontrar el archivo, se establecerá la configuración por defecto y se creará el archivo. Al ser en texto claro, este archivo es modificable con

cualquier editor de texto, aunque no se recomienda.

### 3.2.3.- Interfaz de usuario

El nivel de interfaz de usuario es el nivel encargado de la interacción entre el sistema y el usuario. Es este nivel con el que el usuario va a trabajar, recibiendo información y actuando en consecuencia. La clave de un buen nivel de interfaz radica en exponer la información de manera clara y concisa, organizando esa información en pantallas que se centran en tareas concretas y ofreciendo una navegabilidad predecible y bien estructurada.

En resumen, debe apuntarse a una buena usabilidad de la interfaz. Para conseguir tal usabilidad la interfaz debe permitir realizar la tarea para la que se diseña, poder hacerlo cómodamente y sin consumir demasiado tiempo y ser fácil de aprender.

Es este nivel el encargado de enlazar al usuario con la funcionalidad del sistema, recogiendo los datos de entrada de éste y dirigiéndolos al nivel funcional, mostrando el resultado con la nueva información y los cambios producidos de nuevo al usuario.

#### 3.2.3.a Adaptación al usuario final

Uno de requisitos del sistema prioriza la facilidad de uso de la interfaz frente a una mayor o mejor funcionalidad. El entorno de trabajo y el perfil del usuario final son las razones que llevan a esta decisión. En el entorno a la intemperie y de carácter industrial la necesidad de precisión en el manejo del sistema supone un inconveniente; y el perfil de usuario final, poco familiarizado con los sistemas informáticos limita el uso de controles avanzados y complejos en el sistema.

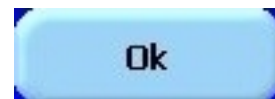
Con el objetivo de reducir la interacción con el sistema a operaciones básicas de hacer clic e introducción de datos en campos, evitando operaciones de arrastre, dobles clics o combinaciones de teclas se establecieron estas guías generales a seguir en toda la interfaz.

- **Botones grandes** : los botones estándar ofrecidos por el entorno de desarrollo son apropiados para la mayoría de aplicaciones, pero en nuestro no se adaptaban a las necesidades de facilidad de uso requeridas. Se diseñaron nuevos botones con un área y tamaño de texto mayor, eliminando la necesidad de emplear herramientas de precisión para activarlos.

Original



Frente a rediseñado



- **Texto grande** : el formato estándar para el texto ofrecido por el entorno de desarrollo es apropiado para aplicaciones comunes que están pensadas para ser utilizadas en entornos con buena claridad o donde puede conseguirse fácilmente una buena claridad. En el entorno de trabajo objetivo del sistema no puede garantizarse una buena claridad, por lo que se ha adaptado todo el texto para presentarse a tamaño grande y con gran contraste con el fondo, procurando facilitar en lo posible la lectura.

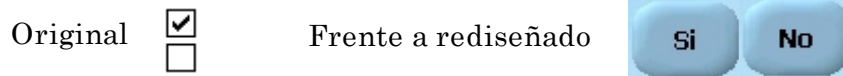
Original

direc 1548; OLIVA

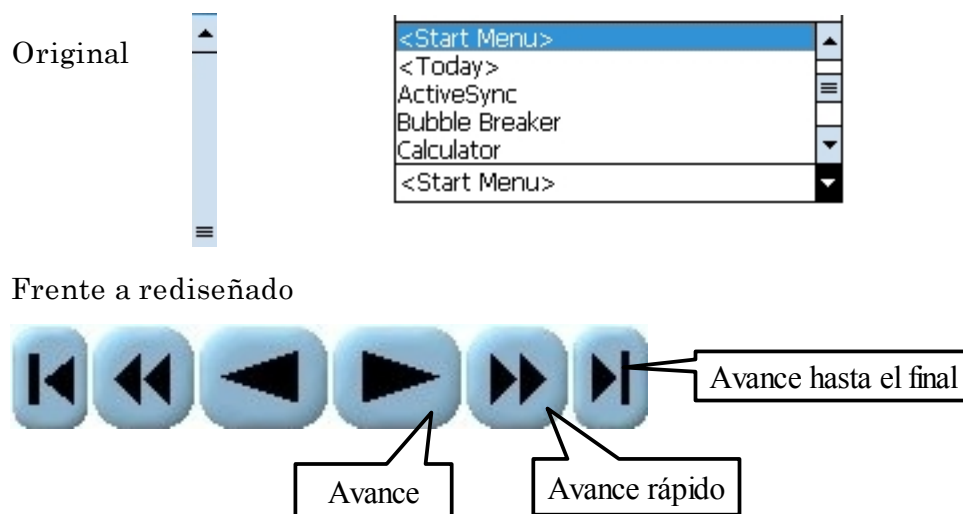
Frente a rediseñado

**direc 1548; OLIVA**

- **Evitar cajas de selección** : el tipo de caja de selección ofrecido en el entorno de desarrollo es demasiado pequeño y no es apropiado para el sistema al requerir de instrumentación de precisión para utilizarlo. Se ha sustituido el caja de selección por botones **Sí/No** con la misma funcionalidad, pero con un área y texto mayor.



- **Evitar listas desplegables y barras de desplazamiento** : estos controles requieren para su correcto uso tanto precisión como una operación que procuramos evitar, el arrastre. Sustituimos pues estos controles por un esquema de navegación en el que, a pantalla completa, se muestran los elementos en pequeños grupos a través de los cuales puede navegarse empleando la barra de navegación.



La selección de un elemento en este esquema no supone más que hacer clic en el elemento deseado.

- **Poca información por pantalla** : cada pantalla mostrará la información necesaria y sólo añadirá información extra si ayuda a obtener una mejor usabilidad sin complicar el manejo. Así para cada pantalla se diseñó primero la información esencial que se requería, con sus controles, y se añadió información o funcionalidad extra cuando no suponía un sacrificio de la sencillez de uso.
- **Minimizar el uso del teclado** : ante la opción de emplear el teclado o el ratón/pluma para una acción, debe prevalecer la opción del ratón/pluma, y tal vez añadir la opción de uso del teclado como alternativa. De esta forma prácticamente toda tarea puede realizarse mediante clics en pantalla. Hasta donde ha sido posible, la entrada de datos se ha limitado a valores numéricos, que son más fáciles de introducir gracias al teclado que incorpora el dispositivo.

123	1	2	3	4	5	6	7	8	9	0	-	=	↩
Tab	q	w	e	r	t	y	u	i	o	p	[	]	
CAP	a	s	d	f	g	h	j	k	l	;	'		
Shift	z	x	c	v	b	n	m	,	.	/		↵	
Ctl	á	ü	`		\							↓	↑

*Ilustración 18: Teclado en pantalla*

### 3.2.3.b Navegación en la interfaz

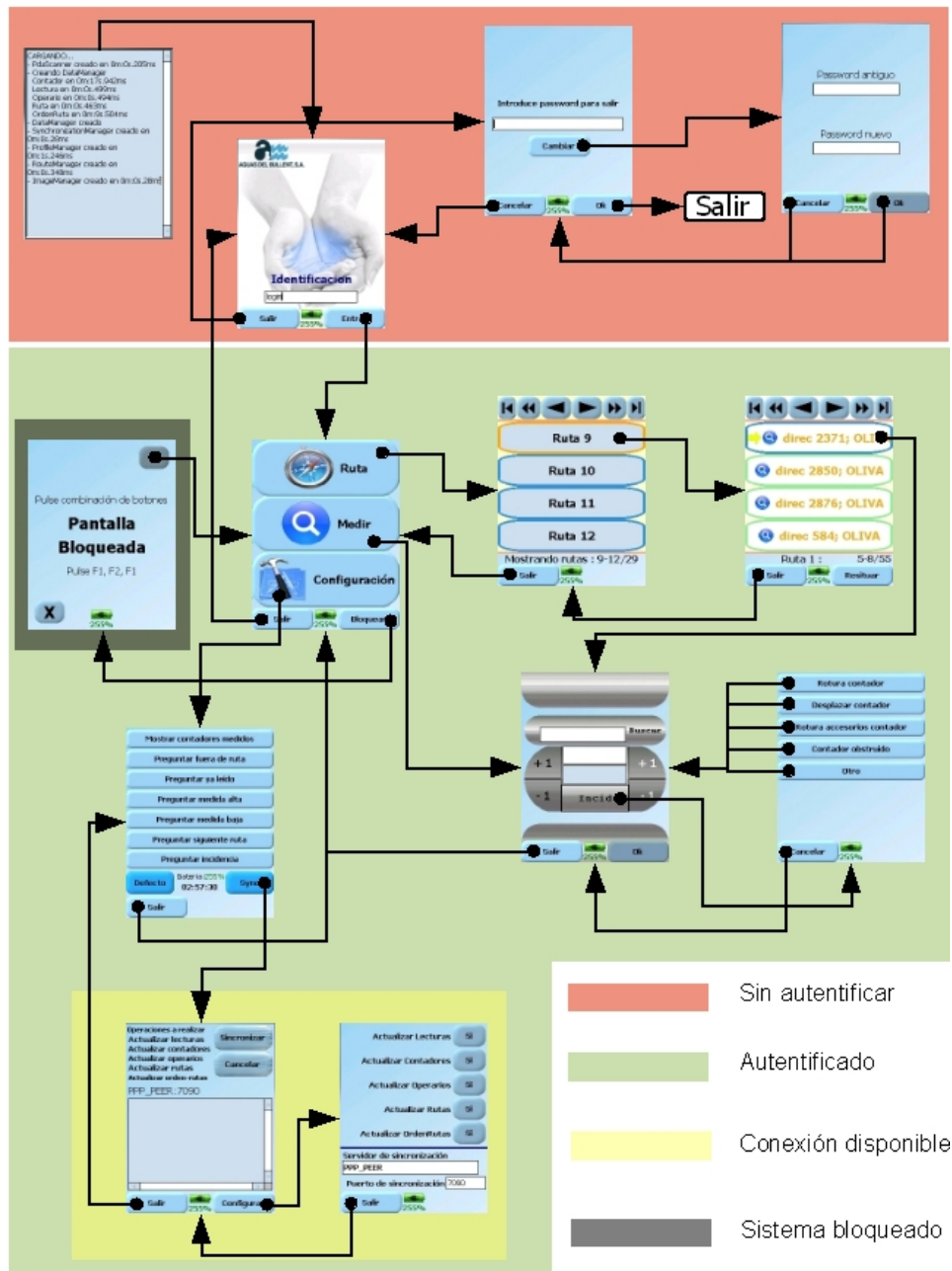


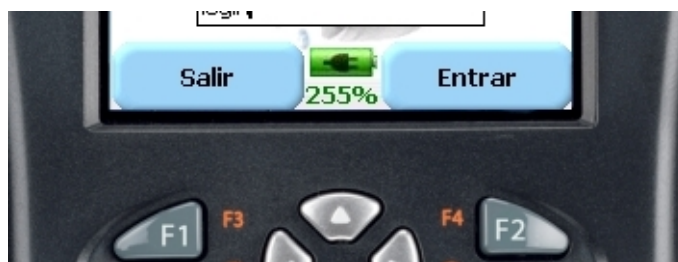
Ilustración 19: Navegación en **waterLU**

Con tal de mantener la sencillez de uso en el manejo de la interfaz, se ha establecido un convenio de control general para todas las pantallas, creando consistencia en el manejo del sistema así como un comportamiento predecible de la interfaz. Se han establecido unas guías comunes a través de todo el sistema ayudando al aprendizaje y a la sencillez de uso.

1) La primera de estas guías corresponde al comportamiento de la tecla **ESC** (escape). Por diseño, pulsar esta tecla en una pantalla de la interfaz provoca que se abandone esa pantalla, con el mismo comportamiento que si se pulsara el botón de salir o cancelar en la pantalla. Si se hubieran producido cambios, se cancelarían.

2) La segunda de estas guías establece el comportamiento de las teclas **F1** y **F2**. Todas las pantallas de la interfaz tiene uno o dos botones en la

parte inferior a modo de opciones de menú. La disposición de las teclas **F1** y **F2** del teclado del dispositivo **Datalogic Blackjet™** y de la mayoría de PDAs invita a hacer correspondientes el pulsar esas teclas con hacer clic en los botones. Es por esto que se ha adaptado la interfaz para que la tecla **F1** se corresponda siempre con hacer clic en el botón inferior izquierdo y la tecla **F2** se corresponda con hacer clic en el botón inferior derecho.



*Ilustración 20: Botones F1 y F2*

3) La última de estas guías trata de dar a la tecla **ENTER** su comportamiento clásico de entrada de datos o de confirmación de selección. Así en un campo de entrada de datos, al pulsar la tecla **ENTER** se realizará la operación a esperar, por ejemplo, identificarse al estar introduciendo la identificación o guardar la lectura al estar en la pantalla de toma de lecturas.

### 3.2.3.c Pantallas funcionalidad

El sistema cuenta con varias pantallas, cada una con un objetivo concreto, aprovechando el poco espacio de pantalla disponible. La mayoría de estas pantallas cumple parte de la funcionalidad especificada en los requisitos, pero existen algunas que aportan una funcionalidad no requerida aunque interesante sin impactar negativamente en la facilidad de uso del sistema.

Se presentan aquí las pantallas que cumplen con la funcionalidad requerida :

- **Identificación** : Encargada de identificar y autenticar al usuario del dispositivo. Es obligatorio pasar por este punto del sistema para poder realizar tareas. Una identificación correcta permite entrar en el sistema con una nueva sesión.

Pulsar **Salir** lleva a **Abandono de la aplicación**.

Pulsar **Entrar** permite entrar en el sistema si la identificación es correcta, lleva a **Menú principal**.



*Ilustración 21: Pantalla identificación*

- **Menú principal** : Organizador del sistema, permite pasar a **Seguimiento de rutas**, pasar directamente a **Toma de lecturas** o pasar a **Configuración general**. La opción de configuración puede no estar disponible si desde el servidor central se ha inhabilitado el cambio de configuración en el dispositivo.



*Ilustración 23: Pantalla menú principal con configuración*



*Ilustración 22: Pantalla menú principal sin configuración*

Abandonar esta pantalla termina la sesión en marcha, y será necesario volver a **identificarse** ante el sistema para entrar de nuevo.

Desde esta pantalla puede también **bloquearse** la aplicación.

- **Selección de ruta** : Pantalla con un esquema de navegación que permite recorrer las rutas existentes de manera fácil y rápida, permitiendo seleccionar una de ellas y comenzar su itinerario. Aplica el esquema de navegación mediante una barra con botones que cambian la selección recorriendo los existentes en saltos de diferente tamaño. Abandonar esta pantalla lleva al **Menú principal**. Seleccionar una de las rutas lleva a la pantalla de **Seguimiento de ruta**.



*Ilustración 24: Pantalla de selección de ruta*



- **Seguimiento de ruta** : Muestra en orden por posiciones en la ruta un resumen del estado de los contadores. Empleando un código de iconos y colores se da información de cada contador. Aplica un esquema de navegación mediante una barra con botones similar al empleado en la pantalla de **Selección de ruta**.

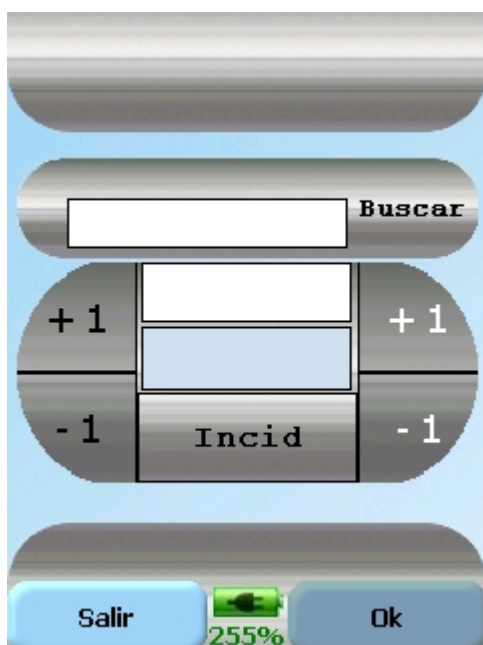
Cada elemento representa un contador, mostrando su dirección. El aspecto determina el estado, de la siguiente forma: un fondo blanco con el icono indica que el contador no ha sido leído; mientras que un fondo gris indica que el contador ya ha sido leído. En los contadores ya leídos, un icono nos informa de que la lectura se tomó correctamente; en cambio un icono informa de que se registró una incidencia durante o en lugar de la lectura.



*Ilustración 25: Pantalla de selección de contador*

Abandonar esta pantalla lleva a la pantalla de **Selección de ruta**. Seleccionar uno de los contadores lleva a la pantalla de toma de lecturas con ese contador preparado para ser leído. Pulsar el botón **Resituar** devuelve la navegación al contador que debería leerse a continuación.

- **Toma de lecturas** : Permite la toma de lecturas. Adaptada y optimizada con el objetivo de realizar las lecturas en el menor tiempo posible, ofrece diversas facilidades para identificar un contador y tomar una lectura.



*Ilustración 26: Pantalla de toma de lecturas sin datos*



*Ilustración 27: Pantalla de toma de lecturas con datos*

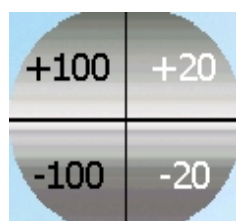
La identificación de un contador en esta pantalla puede realizarse, o bien introduciendo manualmente el código de contador y pulsando **Buscar**, o bien



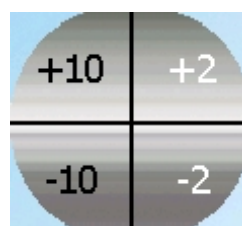
empleando el lector de códigos de barras del dispositivo **Datalogic Blackjet**™, o bien por el seguimiento de una ruta.

Al identificar un contador, se muestra su información y el foco pasa directamente al campo de entrada de la lectura, en el que se muestra el valor previsto de consumo totalmente seleccionado de forma que pueda sobrescribirse con un nuevo valor si se desea sin necesidad de realizar la selección a mano.

Hay cuatro botones con dos valores diferentes que permiten modificar el valor de la lectura sin necesidad de teclear un nuevo valor; esto permite realizar pequeños ajustes de manera rápida. Cabe destacar que los valores de los botones se calculan en función del histórico de consumo del contador, por lo que cuanto mayor sea el consumo del contador mayores serán los valores.



*Ilustración 28:  
Ajustador con  
valores 100 y 20*

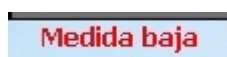


*Ilustración 29:  
Ajustador con  
valores 10 y 2*

Llegar a esta pantalla mediante la selección de un contador en una ruta la prepara con los datos del contador. La información del contador nos permite mostrar una previsión del consumo en el campo de entrada de la lectura. A modo de referencia se muestra también la última lectura que se realizó del contador. Los análisis del histórico de consumo del contador nos permiten, además de calcular una previsión del consumo, definir un intervalo de valores que marcan el rango en que debe estar comprendida la lectura para no ser considerada anómala.

En caso de estar siguiendo una ruta, en la parte superior se muestra información de la posición del contador en la ruta, y en la inferior información del siguiente contador en la ruta.

Pulsar el botón **OK** guarda la lectura para el contador, y en caso de seguir una ruta, pasa al siguiente contador por medir de la ruta. Dependiendo de la configuración establecida en el sistema, puede que no se permita guardar una lectura por encontrarse fuera del rango de valores de consumo normales para este contador. De ser así se avisa en la pantalla, debajo del campo introducción de lecturas.



*Ilustración 30:  
Aviso medida  
baja*



*Ilustración 31:  
Aviso medida  
alta*

En caso de estar tomando una lectura de un contador que ya ha sido leído, se informa de ello en la pantalla, encima del número de contador.

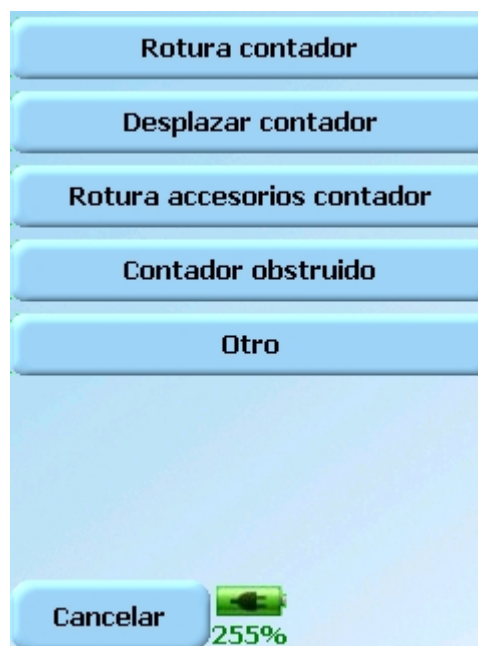
*Ilustración  
32: Aviso  
de relectura*

El registro de una incidencia, o se realiza automáticamente por el sistema para lecturas con valores anómalos, o se realiza mediante la pantalla de incidencias pulsando **Incid.**

Pulsar **Salir** abandona esta pantalla llevando al **Menú principal**.

- **Registro de incidencias** : Permite registrar la incidencias no detectables automáticamente por el sistema. Pulsando sobre una de las descripciones de incidencia se registra una incidencia del tipo adecuado con un mensaje detallado y específico para el contador afectado. Un contador con una incidencia registrada se considera ya leído.

Pulsar **Cancelar** lleva a la pantalla de **Toma de lecturas** cancelando el registro de la incidencia.



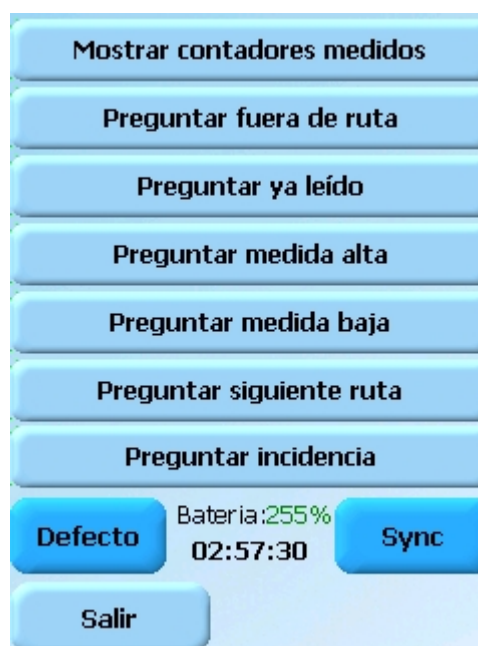
*Ilustración 33: Pantalla de registro de incidencias*

- **Configuración general** : En esta pantalla se muestran opciones generales de configuración que determinarán el comportamiento del sistema, permitiendo, prohibiendo o preguntando ante determinadas acciones del usuario. Esta pantalla puede estar inhabilitada si desde el servidor central no se permite el cambio de la configuración del dispositivo.

Pulsando sobre opción se cambia su valor, rotando entre todos los valores posibles.

En cualquier momento puede volverse a la configuración por defecto pulsando el botón **Defecto**.

El botón **Sync** lleva a la pantalla de sincronización, y estará disponible



*Ilustración 34: Pantalla de configuración general*

únicamente si el dispositivo dispone de conexión.

Pulsar **Salir** lleva a **Menú principal**.

- **Sincronización** : Desde esta pantalla se realiza la sincronización de datos entre el dispositivo y el servidor central. Muestra información de las operaciones a realizar así como contra que servidor se realizarán.

Pulsar **Sincronizar** inicia la sincronización. Mientras dure la sincronización y actualización de datos se mostrará el progreso del proceso en el cuadro de texto.

Pulsar **Cancelar** detendrá el proceso de sincronización y anulará cualquier cambio en los datos.

Pulsar **Salir** abandona la pantalla llevando a la **Configuración general** o a **Menú principal** si la configuración está inhabilitada.

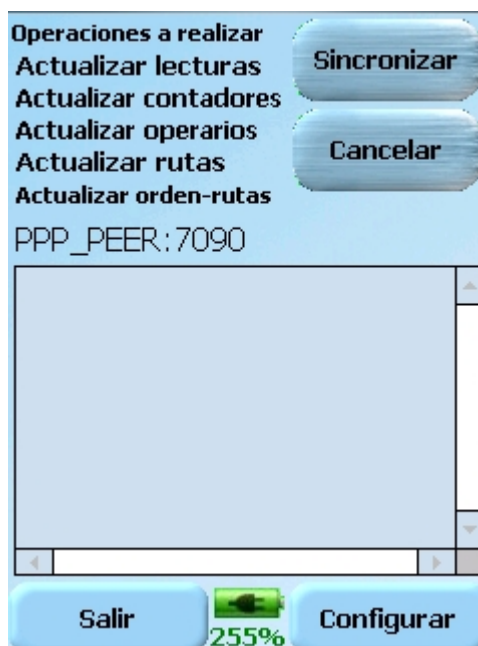
Pulsar **Configurar** lleva a **Configuración de sincronización**, donde puede configurarse el proceso de sincronización.

- **Configuración de sincronización** : Esta pantalla permite especificar los parámetros del proceso de sincronización, esto es, decidir que operaciones se realizarán y contra que servidor y puerto se llevará a cabo el proceso.

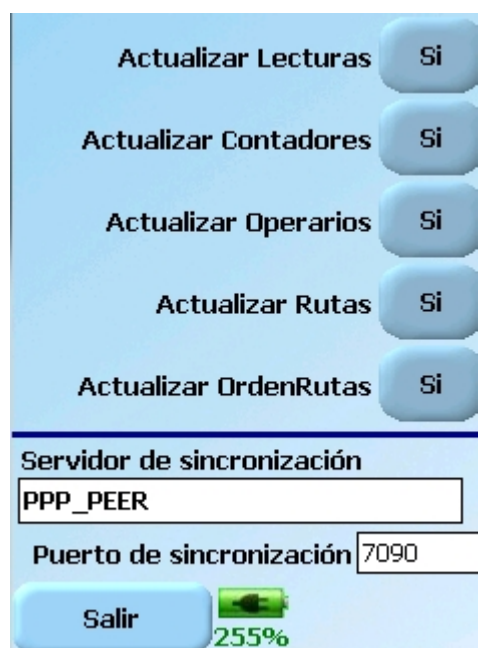
Pulsando sobre el botón contiguo a una operación puede seleccionarse si se realizará o no esa operación, alternando entre las opciones **Sí/No**.

En los campos Servidor de sincronización y puerto de sincronización puede especificarse el nombre **DNS** o **IP** y el puerto TCP donde se encuentra el servidor contra el que se realizará la sincronización.

Pulsar **Salir** lleva a **Sincronización**.



*Ilustración 35: Pantalla de sincronización*



*Ilustración 36: Pantalla de configuración de sincronización*

### 3.2.3.d Pantallas especiales

Estas pantallas añaden funcionalidad no especificada en los requisitos. Se han añadido por la utilidad que presenta la funcionalidad que implementan sin interferir en la sencillez de uso del sistema. La mayoría de estas pantallas se emplearan muy poco, aún así, en el momento en que son necesarias pueden resultar muy útiles.

- **Cargando** : Pantalla que al iniciar la aplicación muestra información del proceso de inicialización y carga de datos desde la base de datos del dispositivo. El proceso de carga de la aplicación puede llevar desde aproximadamente 30 segundos a 2 minutos, dependiendo del volumen de información en la base de datos, la memoria y procesador del dispositivo, el sistema operativo que corre el dispositivo, etc.

Es por el dilatado espacio de tiempo que requiere la carga del sistema que se diseñó y añadió la pantalla de carga, en la que se muestra el progreso de inicialización de las diferentes capas que forman el sistema. Esta información también se guarda en un fichero de log.

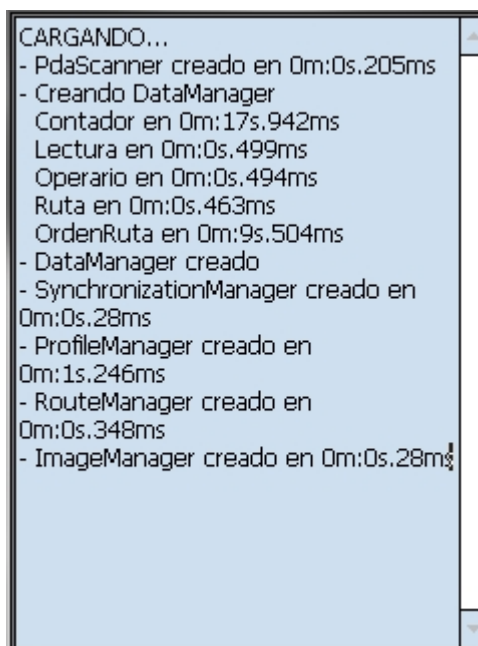
Una vez completado el proceso de carga, se muestra la pantalla **Identificación**.

- **Abandonando la aplicación** : El sistema diseñado no debe poder abandonarse sin permiso, por lo que al intentar abandonar la aplicación se pide una contraseña para completar el abandono. Esta pantalla se encarga de pedir la contraseña, verificarla y permitir o denegar al abandono según la contraseña introducida.

Pulsar **Cancelar** lleva a **Identificación**.

Pulsar **Cambiar** lleva a **Cambiar contraseña de abandono**.

Pulsar **Ok** abandona la aplicación siempre que la contraseña introducida sea la correcta.



*Ilustración 37: Pantalla de carga*



*Ilustración 38: Pantalla de abandono de la aplicación*

- **Cambiar contraseña de abandono :** Esta pantalla tiene como propósito permitir cambiar la contraseña de abandono de la aplicación. Con tal de cambiar la contraseña con éxito es necesario conocer la antigua contraseña e introducir una contraseña nueva con al menos un carácter alfanumérico.

Pulsar **Cancelar** vuelve a la pantalla de **Abandono de la aplicación**.

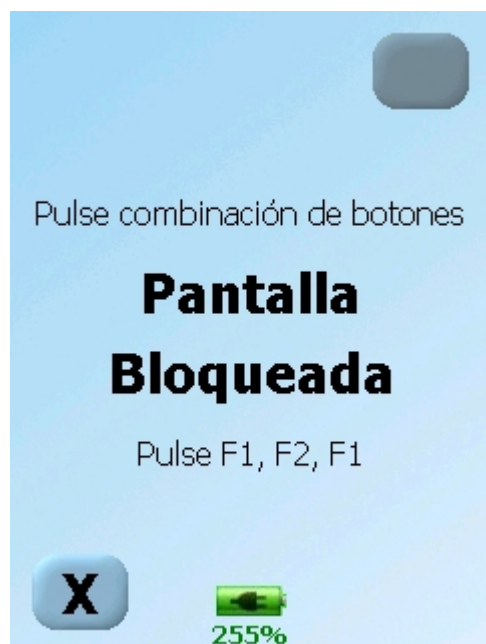
Pulsar **Ok** cambiará la contraseña antigua por la nueva introducida. Cabe destacar que el botón **Ok** no estará habilitado hasta haber introducido correctamente la antigua contraseña.

*Ilustración 39: Pantalla de cambio de contraseña*

- **Bloqueo del sistema :** Esta pantalla bloquea el sistema, evitando que puedan realizarse operaciones o cambios accidentales y no deseados. Mientras el sistema se encuentre en esta pantalla hará caso omiso de cualquier pulsación de tecla o en la pantalla, excepto de aquellas que permiten abandonar la pantalla.

Para abandonar esta pantalla y volver al **Menú principal** es necesario pulsar una combinación de teclas, ya sea en teclado o en pantalla, que es muy poco probable que se den accidentalmente.

Así puede abandonarse la pantalla pulsando los botones en el orden adecuado, o bien pulsando la combinación de teclas **F1-F2-F1**.



*Ilustración 40: Pantalla de bloqueo*

- **Mensajes de confirmación y aviso :** Durante la ejecución de tareas con los dispositivos, van a presentarse situaciones y acciones que podríamos calificar de anómalas, entre ellas y sin limitarnos a : medición de contadores fuera de la ruta seguida, toma de lecturas muy altas o bajas, medición de contadores ya medidos, etc.

Ante estas situaciones el sistema debe decidir como reaccionar. El comportamiento a seguir está definido en la capa de configuración, donde se ha establecido entre las opciones posibles cuál es la que debe tomar el



sistema.

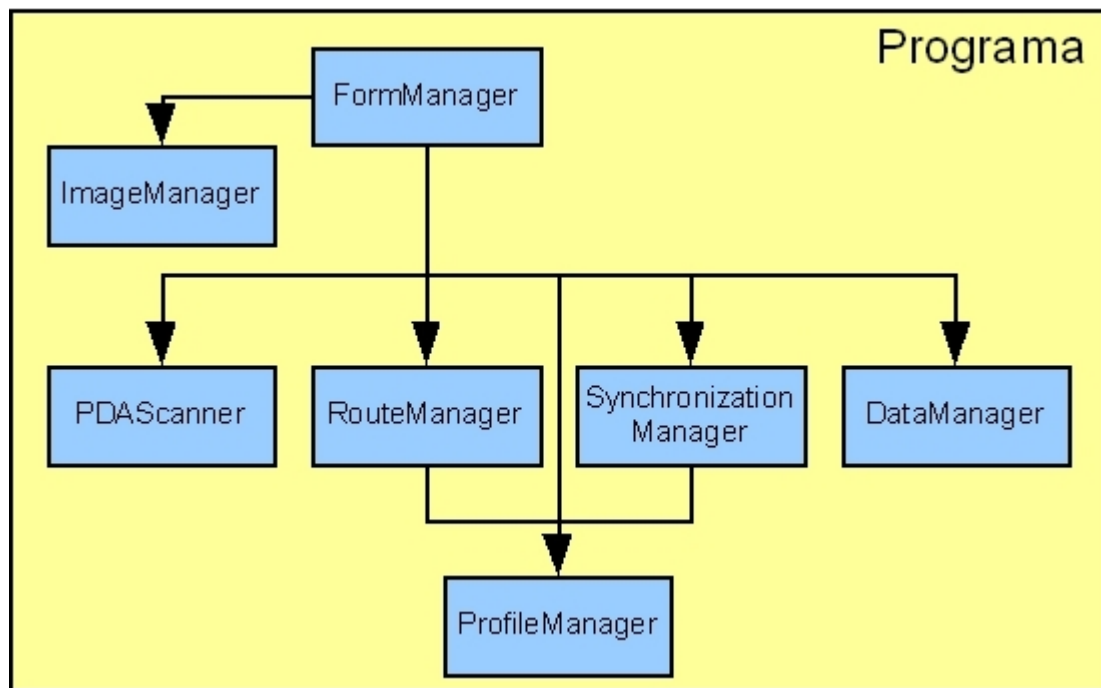
Así nos encontraremos que al presentarse una de estas situaciones anómalas el sistema puede preguntarnos como proceder o puede avisarnos de que esa acción no está permitida. Tanto para pedir nuestra confirmación como para informarnos de la imposibilidad de llevar a cabo un acción, el sistema empleará mensajes de confirmación y aviso.

### 3.2.4.- Gestores de la aplicación

La aplicación recurre a diversos gestores que, encargándose cada uno de una tarea específica, en conjunto proporcionan los tres niveles de datos, lógica de negocio e interfaz de usuario. El programa conoce y tiene referencia a todos esto gestores, y algunos de ellos emplean y tienen referencias a otros gestores, formándose una jerarquía en la que unos gestores dependen o emplean a otros.

La división de la aplicación en diferentes gestores ayuda a la reutilización de código y a la estructuración del desarrollo. Manteniendo el código agrupado por funcionalidad mediante los gestores permite también que en el futuro el mantenimiento se lleve a cabo de forma más rápida y sencilla, al encontrar cada funcionalidad centralizada en un punto.

Los gestores desarrollados en esta aplicación, y su interacción :



*Ilustración 41: Gestores y su dependencia del programa*

Junto con los gestores la aplicación desarrollada se ayuda de un estado de la aplicación que determina a que estados puede pasar y que operaciones pueden realizarse. El estado de la aplicación coincide con la navegación por las diferentes pantallas que forman la aplicación, de forma que cada estado se corresponde con una o varias de las pantallas de la aplicación.

La funcionalidad que aporta cada gestor, y el papel que juega en la aplicación son :

### 3.2.4.a ProfileManager

Gestor de perfiles y configuración. Encargado de las sesiones y de la configuración. Mediante las sesiones controlamos que operario está realizando tareas con el dispositivo, y que no haya más que un operario en cada momento por dispositivo.

La gestión de la configuración se centra en tres aspectos :

- Cargar y guardar la configuración desde y en fichero, de forma que los cambios en la configuración estén disponibles de una a otra ejecución de la aplicación.
- Controlar el comportamiento de la aplicación ante determinadas acciones anómalas por parte del usuario mediante parámetros para esas acciones que tendrán uno de estos valores : **Allow**, **Deny** o **Prompt** (**Permitir**, **Prohibir** o **Preguntar**).
- Habilitar o inhabilitar los cambios en la configuración de forma local al dispositivo. Desde **AguasCli** puede establecerse una configuración para la aplicación del dispositivo y una de las opciones permite inhabilitar los cambios de configuración en el dispositivo.

Ofrece públicamente estas propiedades y métodos que determinan el comportamiento y otros factores de la aplicación :

- Para el control de sesiones :
  - **IDOperario** : el identificador del operario que ha iniciado sesión en la aplicación. Será cadena vacía si no hay una sesión abierta.
  - **UserIdentified** : **TRUE** cuando haya una sesión abierta, **FALSE** en caso contrario.
  - **NewSession()** : Crea una nueva sesión. Fallará si ya existía una sesión creada.
  - **CloseSession()** : Cierra la sesión en curso.
- Para el control de situaciones anómalas :
  - **ShowMeasuredContador** : Determina si deben o no mostrarse los contadores ya vistos en la pantalla de [Seguimiento de ruta](#).
  - **AskOutOfRoute** : Determina el comportamiento al intentar medir un contador que no es el siguiente en la ruta.
  - **AskAlreadyMeasured** : Determina el comportamiento al intentar medir un contador ya medido.
  - **AskTooHighMeasured** : Determina el comportamiento al intentar introducir una lectura demasiado alta respecto a la esperada.
  - **AskTooLowMeasure** : Determina el comportamiento al intentar introducir una lectura demasiado baja respecto de la esperada.
  - **GoToNextRoute** : Determina el comportamiento al leer el último contador de una ruta.
  - **AskIncidence** : Determina el comportamiento al intentar introducir una incidencia para un contador.
  - **GoToMeasureAfterScanner** : Determina el comportamiento tras leer un código de barras mediante el escáner.

- Para el control de sincronización :
  - **UpdateLecturas** : Determina si se llevará a cabo la operación de registro de lecturas e incidencias.
  - **UpdateContadores** : Determina si se llevará a cabo la operación de actualización de contadores.
  - **UpdateOperarios** : Determina si se llevará a cabo la operación actualización de operarios.
  - **UpdateRutas** : Determina si se llevará a cabo la operación de actualización de rutas.
  - **UpdateOrdenRutas** : Determina si se llevará a cabo la operación actualización de OrdenRutas.
  - **SynchronizationServer** : Proporciona el nombre **DNS** o **IP** donde estará escuchando **AguasCli** esperando la sincronización.
  - **SynchronizationPort** : Proporciona el puerto donde estará escuchando **AguasCli** esperando la sincronización.
- Para el control de cambios de la configuración :
  - **SobreescribirConfigPDA** : Indica si debe sobrescribirse la configuración actual con la recibida desde **AguasCli**.
  - **PermitirCambioConfigLocalPDA** : Indica si los cambios locales en el dispositivo están habilitados.
  - **LoadPreferences()** : Carga la configuración de la aplicación desde el fichero de configuración. A falta de este archivo se crea una configuración por defecto.
  - **SavePreferences()** : Guarda la configuración actual de la aplicación en el fichero de configuración.
  - **AcceptChanges()** : Acepta los cambios realizados a la configuración desde el último **AcceptChanges()** o **RejectChanges()**.
  - **RejectChanges()** : Rechaza los cambios a la configuración desde el último **AcceptChanges()** o **RejectChanges()**.
- Para el control de contraseñas :
  - **CheckLeavingPassword()** : Compara una cadena con la contraseña establecida en la configuración.
  - **SetLeavingPassword()** : Establece una cadena como la nueva contraseña para abandonar la aplicación.

#### **3.2.4.b PDAScanner**

Gestor del escáner del dispositivo. Encargado de detectar la presencia de un escáner en el dispositivo, y en su caso ponerlo a disposición del programa. Proporciona funciones de inicialización y terminación del escáner, eventos lanzados al realizarse una lectura correcta a los que puede suscribirse delegados, y una lectura de los datos leídos como entero o como cadena.

Estos son las propiedades y métodos que pone a disposición pública :

- Para datos y disponibilidad :



- **Data** : Proporciona los datos leídos tras una lectura con el escáner en forma de cadena (**string**).
- **NumContador** : Proporciona los datos leídos tras una lectura con el escáner como un entero (**integer**) representando un código de contador.
- **IsAvailable** : Indica si el dispositivo móvil dispone de escáner o no.
- Para inicialización y terminación :
  - **InitScanner()** : Inicializa y habilita el escáner para poder realizar lecturas con él.
  - **DeInitScanner()** : Inhabilita y termina el escáner.
- Para disparos del escáner por software :
  - **StartShot()** : Inicia un disparo del láser del escáner por software. El disparo deberá pararse por software.
  - **StopShot()** : Detiene el disparo del láser del escáner por software.
  - **Shot()** : Inicia un disparo por software, duerme n milisegundos y detiene el disparo por software, de forma que el disparo está activo n milisegundos para leer.
- Eventos de datos leídos:
  - **DatoLeído** : Evento lanzado al leer correctamente un código de barras con el escáner. Antes de lanzar este evento se han establecido los valores de las propiedades **Data** y **NumContador** con el valor del código leído.

#### 3.2.4.c *RouteManager*

Gestor de rutas. Encargado de dar soporte al seguimiento de rutas. Desde el inicio de la aplicación carga las rutas disponibles permitiendo navegar por ellas. Una ruta no cargará los contadores que la componen hasta que sea solicitada, minimizando el consumo de memoria.

Ofrece métodos para la navegación por las rutas y sus contadores, permitiendo avanzar o retroceder de uno en uno o en otros incrementos. Opcionalmente puede establecerse la discriminación de los contadores ya visitados de forma que se ocultan para la navegación.

Ofrece estas propiedades métodos con el fin de proporcionar la funcionalidad de seguimiento de rutas:

- De información general de rutas :
  - **RouteFollowed** : Proporciona la ruta que se está siguiendo.
  - **AllRoutes** : Proporciona una lista con todas las rutas.
  - **PrevToShow** : Indica en la navegación si hay rutas anteriores a las seleccionadas actualmente.
  - **NextToShow** : Indica en la navegación si hay rutas posteriores a las seleccionadas actualmente.
  - **ShowingInfo** : Proporciona una cadena con información de la ruta seguida y su orden entre todas las rutas.
- Operaciones de actualización :
  - **AddNewRoute()** : Añade una nueva ruta al gestor. Estará disponible

desde este mismo instante.

- **DeleteRoute()** : Borra una ruta del gestor. Dejará de estar disponible desde este mismo instante. Si ésta era la ruta que se estaba siguiendo pasaremos a no seguir ninguna ruta.
- **UpdateRoute()** : Actualiza una ruta del gestor. Los cambios están disponibles desde este mismo instante.
- **AddNewOrdenRuta()** : Añade un contador en una ruta con el orden indicado. El contador estará disponible en la ruta desde este mismo instante.
- **DeleteOrdenRuta()** : Borra un contador de una ruta. El contador dejará de estar disponible en la ruta desde este mismo instante.
- **UpdateOrdenRuta()** : Actualiza el orden de un contador en una ruta. El nuevo orden del contador se hará efectivo en este mismo instante.
- Seguimiento y navegación por rutas :
  - **FollowRoute()** : Establece la ruta pasada como la ruta a seguir. Pasamos a seguir esa ruta.
  - **UnFollowRoute()** : Deja de seguir la ruta actual.
  - **GetBlock()** : Devuelve la selección de rutas según indica el puntero de navegación.
  - **GetNextBlock()** : Devuelve la selección de rutas avanzando el puntero de navegación.
  - **GetPrevBlock()** : Devuelve la selección de rutas retrasando el puntero de navegación.
  - **MeasureNextContador()** : Pasa a medir el siguiente contador en la ruta seguida.
  - **MeasureContador()** : Pasa a medir el contador indicado. Si ese contador no está en la ruta seguida dejamos de seguir esa ruta.
  - **GoToNextToNextRoute()** : Pasa a medir la ruta siguiente a la actualmente seguida.

#### **3.2.4.d DataManager**

Gestor de acceso a datos. Encargado de ofrecer un acceso, modificación o borrado claro para los datos. Apoyándose en las estructuras **ADO.NET** proporciona un acceso sencillo a las tablas y relaciones del modelo de datos, que en nuestro caso servirá SQL Server 2005 Mobile. La búsqueda de datos puede realizarse mediante las claves primarias u otros campos, flexibilizando el acceso a los datos.

Proporciona métodos para la actualización e inserción de los datos, haciendo transparente todo el proceso de validación e introducción en la base de datos :

- Tablas de datos :
  - **ContadorTable** : Tabla de contadores, contendrá todos los contadores conocidos por la aplicación desde la última actualización.
  - **LecturaActualTable** : Tabla de lecturas e incidencias, contendrá todas las lecturas e incidencias tomadas desde la última sincronización.

- **OperarioTable** : Tabla de operarios que pueden autenticarse en la aplicación, contendrá los operarios conocidos por la aplicación desde la última sincronización.
- **RutaTable** : Tabla de rutas conocidas por la aplicación, contendrá todas las rutas conocidas desde la última sincronización.
- **OrdenRutasTable** : Tabla de ordenes de contadores en rutas conocidos por la aplicación, contendrá los órdenes de los contadores conocidos desde la última sincronización.
- Inserciones y actualizaciones de datos :
  - **InsertLectura()** : Inserta una lectura o incidencia en la tabla **LecturaActual** que guarda las lecturas e incidencias, actualizando las relaciones y claves ajenas.
  - **InsertContador()** : Añade un contador a la tabla de contadores
  - **InsertOperario()** : Añade un operario a la tabla de operarios.
  - **InsertRuta()** : Añade una ruta a la tabla de rutas.
  - **InsertOrdenRutas()** : Añade un orden de un contador en una ruta, actualizando las relaciones y claves ajenas.
  - **UpdateLectura()** : Actualiza una lectura o incidencia de la tabla de lecturas e incidencias.
  - **UpdateContador()** : Actualiza un contador de la tabla de contadores..
  - **UpdateOperario()** : Actualiza un operario de la tabla de operarios.
  - **UpdateRuta()** : Actualiza una ruta de la tabla de rutas.
  - **UpdateOrdenRutas()** : Actualiza el orden de un contador en una ruta en la tabla de órdenes de ruta.
- Cambios en tablas :
  - Operaciones de actualización : **Update\*Table()** : Para cada una de las tablas de datos (contadores, lecturas, operarios, rutas y ordenes de ruta) se ofrece un método que actualiza la tabla de la base de datos tomando como referencia la tabla de que dispone la aplicación en memoria.
  - Operaciones de refresco : **Refresh\*()** : Para cada una de las tablas de datos se ofrece un método que vacía las tablas de datos en memoria y vuelve a rellenarlas con los datos de la base de datos.
  - Operaciones de descarte de cambios: **Reject\*TableChanges()** : Para cada una de las tablas de datos se ofrece un método que permite descartar los cambios realizados a la tabla desde el último **Update\*Table()** o **Reject\*TableChanges()**.

#### 3.2.4.e SynchronizationManager

Gestor de sincronización. Encargado de gestionar el proceso de [sincronización](#), desde el establecimiento de conexión hasta la actualización de base de datos. Emplea un esquema cliente servidor donde el servidor sirve operaciones de actualización de los datos para los dispositivos.

Proporciona métodos para iniciar y detener la sincronización con el servidor, así como eventos a los que suscribir delegados para los eventos de :

- Propiedades y métodos de sincronización :
  - **SynchronizationAvailable** : Indica si la aplicación dispone de conexión con la que realizar una sincronización.
  - **Synchronizing** : Indica si el proceso de sincronización está en marcha.
  - **DefaultHost** : Nombre **DNS** o **IP** del servidor de sincronización por defecto.
  - **DefaultPort** : Número de puerto del servidor de sincronización por defecto.
  - **PerformSynchronization()** : Inicia el proceso de sincronización.
  - **AbortSynchronization()** : Detiene de forma controlada el proceso de sincronización antes de que haya terminado. Produce el rechazo de los cambios recibidos antes de la cancelación.
  - **Reset()** : Reinicia el canal de comunicación que emplea la sincronización.
  - **GetOperationsFromPreferences()** : Obtiene una lista de las operaciones a realizar según está establecido en la configuración de la aplicación.
  - **OperationsDone** : Lista de operaciones que acumula las operaciones ya terminadas durante un proceso de sincronización.
- Eventos lanzados durante una sincronización :
  - Inicio de la sincronización (**Start**) : El proceso de sincronización se ha iniciado, sin haber realizado aún la conexión con [AguasCli](#).
  - Fin de la sincronización (**End**) : El proceso de sincronización ha terminado correctamente, habiendo desconectado de [AguasCli](#) y habiendo actualizado la base de datos.
  - Conexión con [AguasCli](#) (**Connect**) : Se ha establecido la conexión con [AguasCli](#) correctamente; pasaremos a realizar las operaciones.
  - Desconexión con [AguasCli](#) (**Disconnect**) : Se ha desconectado de [AguasCli](#) correctamente; habiendo realizado las operaciones.
  - Inicio de operación (**OperationStarted**) : Se ha iniciado una operación con [AguasCli](#) preparado para atenderla.
  - Avance de operación (**OperationAdvanced**) : La operación en curso ha avanzado o pasado a otra fase.
  - Fin de operación (**OperationFinished**) : Se ha finalizado la operación en curso correctamente.
  - Fallo (**Failure**) : Se ha producido un fallo durante el proceso de sincronización.

#### **3.2.4.f ImageManager**

Gestor de imágenes. Encargado de proporcionar imágenes a la aplicación, especialmente al gestor de formularios.

Ofrece las imágenes necesarias por los formularios y el programa mediante propiedades que encapsulan el formato y nombre de fichero de las imágenes. Permite compartir las imágenes en varios formularios o en el programa sin

necesidad de tener duplicada la imagen en memoria.

Para cada imagen que necesita la aplicación existe una propiedad que al ser solicitada cargará la imagen en caso de no haber sido cargada aún. De esta forma una imagen no ocupará espacio en memoria hasta que sea requerida por la aplicación.

### 3.2.4.g *FormManager*

Gestor de formularios. Encargado de manejar los formularios de la aplicación. Se ofrecen públicamente los formularios, así como métodos públicos para esconder o mostrar formularios.

La gestión de las pantallas y formularios se realiza internamente en el gestor. De esta forma, dependiendo del estado en que se encuentre la aplicación y de la pantalla que se esté mostrando puede cambiar el aspecto de una pantalla o mostrar otra pantalla diferente con el fin de mantener la consistencia en el estado de la aplicación.

Encapsula y centraliza también la gestión de fuentes en la aplicación. Empleando propiedades se ofrecen instancias de fuentes compartidas en toda la aplicación, ahorrando memoria.

Ofrece y proporciona estos métodos y propiedades para la manipulación de la interfaz :

- Para el tratamiento de fuentes :
  - **[ Arial | Courier | Tahoma ] [10 | 12 | 14 | 20] [Regular | Bold]** : Para cada combinación de fuente, tamaño y tipo se ofrece una propiedad que proporciona una instancia de la fuente deseada. La fuente se crea al ser requerida, sin ocupar memoria si no es necesario.
- Pantallas de la aplicación :
  - **Login** : Pantalla de [identificación](#), entrada en el sistema.
  - **Select** : Pantalla de [menú principal](#).
  - **SelectRoute** : Pantalla de [selección de ruta](#).
  - **Route** : Pantalla de [navegación en ruta](#).
  - **Measure** : Pantalla de [toma de lecturas](#).
  - **Incidentcing** : Pantalla de [registro de incidencias](#).
  - **Configure** : Pantalla de [configuración](#).
  - **Synchronization** : Pantalla de [sincronización](#).
  - **SynchronizationConfiguration** : Pantalla de [configuración de sincronización](#).
  - **MessageBox** : Pantalla de [mensajes de alerta y confirmación](#).
  - **Blocking** : Pantalla de [bloqueo de la aplicación](#).
  - **Exiting** : Pantalla de [abandono de la aplicación](#).
  - **PasswordChanging** : Pantalla de [cambio de la contraseña de abandono de la aplicación](#).
  - **Background** : Pantalla trasera empleada como trasfondo del resto de pantallas.

- Métodos y propiedades para el manejo de la interfaz :
  - **CurrentForm** : Hace pública la pantalla presentada actualmente.
  - **ShowForm()** : Presenta al usuario la pantalla especificada.
  - **HideForm()** : Esconde la pantalla indicada.
  - **Show\*()** : Para cada una de las pantallas ofrecidas mediante las propiedades anteriores existe un método que muestra esa pantalla, preocupándose de mantener la consistencia entre aspecto mostrado y estado de la aplicación.

### **3.3.- Entorno de desarrollo**

La aplicación y base de datos han sido desarrolladas sobre la plataforma **.Net Compact Framework 2.0**, empleando la herramienta **Visual Studio 2005** [11] y el lenguaje **C#**.

La plataforma **.Net Compact Framework** ofrece una máquina virtual que abstrae el dispositivo o sistema operativo sobre el que está ejecutándose, permitiendo un diseño y desarrollo sin la necesidad de tener en cuenta detalles de bajo nivel.

La herramienta **Visual Studio 2005** permite integrar perfectamente el desarrollo para la plataforma **.Net Compact Framework** con la interacción con dispositivos, proporcionando un entorno de desarrollo y depuración de muy fácil uso y elevado potencial.

Empleando estas herramientas conseguimos un entorno de desarrollo más cómodo y productivo.

## CAPÍTULO 4- Aplicación OLIVANova

En este capítulo se describe la parte de diseño y desarrollo de la aparte del sistema que añade nuevas funcionalidades al la aplicación ya existente [AguasCli](#). Son descritas las nuevas funcionalidades y como han sido integradas en el código generado automáticamente.

El desarrollo del nuevo sistema de toma de lecturas de contadores, y la implementación de las funcionalidades requeridas obligaba a la inclusión de nuevas funcionalidades en el sistema ya existente; en concreto en la parte cliente que corre sobre un ordenador de sobremesa.

El sistema ya existente ha sido diseñado y generado mediante metodología OLIVANova, como se ha comentado anteriormente. Esta metodología no está aún preparada para ofrecer la funcionalidad que requiere la funcionalidad que se pretende añadir, por lo que será necesario diseñar e implementar a mano las características a incluir.

No obstante, aunque no podamos emplear OLIVANova para diseñar y desarrollar las nuevas funcionalidades por entero, nos apoyaremos en la metodología, clases y servicios que ofrece para implementarlo. Es por esta razón que desde un principio se ha planteado un desarrollo que expande el sistema ya existente añadiendo las funcionalidades a [AguasCli](#), en lugar de intentar desarrollar una aplicación separada que realice las tareas. Es más, integrándolo todo en [AguasCli](#) conseguimos una mejor usabilidad, manteniendo la estructura existente hasta ahora; y permitimos al sistema seguir evolucionando con las nuevas iteraciones del modelo.

### 4.1.- Nuevas funcionalidades añadidas

Con el nuevo proceso de toma de lecturas, el sistema existente debe añadir dos nuevas funcionalidades. Las funcionalidades se añadirán en la parte cliente del sistema, [AguasCli](#), que se ejecuta en un ordenador de sobremesa, ya que es perfectamente capaz de adoptar estas funcionalidades y evitamos así cargar más al servidor central.

Estas dos nuevas funcionalidades son :

1. el proceso de sincronización y
2. la impresión de los códigos de barras.

El proceso de sincronización involucra tanto al dispositivo, al cliente, [AguasCli](#), y al servidor, [AguasSrv](#); y se ha desarrollado enteramente ajeno al modelo que define el sistema, tanto por falta de expresividad de la metodología de modelado como para conseguir una eficiencia mínima.

El mecanismo de impresión de códigos de barras ha sido incluido en el modelo del sistema, aunque la implementación del servicio se ha desarrollado fuera del lenguaje de modelado, tanto por no ser posible expresar su funcionalidad con el lenguaje de modelado como para mantener la ejecución del servicio en el ordenador de sobremesa que corre [AguasCli](#).

#### 4.1.1.- Sincronización

El proceso de **sincronización** es vital. Permite mantener los dispositivos actualizados con los últimos datos encontrados en el servidor, además de registrar las lecturas tomadas y las incidencias ocurridas durante una o más sesiones de tomas de lecturas de contadores.

La parte del proceso de sincronización ejecutada sobre **AguasCli** requiere la interacción del usuario en algunos pasos, definidos en los requisitos.

La primera interacción implica permitir o no realizar la sincronización, pues se requiere que el usuario/operario acepte o cancele la petición de un dispositivo de iniciar una sincronización. Al intentar iniciar una sincronización uno de los dispositivos, en **AguasCli** se pide confirmación del usuario/operario para proceder a llevar a cabo el proceso. Si no se acepta la petición, no se realizará la sincronización y el dispositivo recibirá un mensaje de error.



*Ilustración 42: Confirmación de inicio de sincronización*

La segunda interacción es requerida para precisar la fecha a la que se corresponden las lecturas e incidencias que van a descargarse sobre el servidor central. La facturación de las lecturas se realiza por trimestres o semestres predefinidos, y, según los requisitos definidos, antes de registrar las lecturas o incidencias durante un proceso de sincronización debe permitirse seleccionar la fecha a la pertenecen esas lecturas e incidencias. Se registrarán con esa fecha.



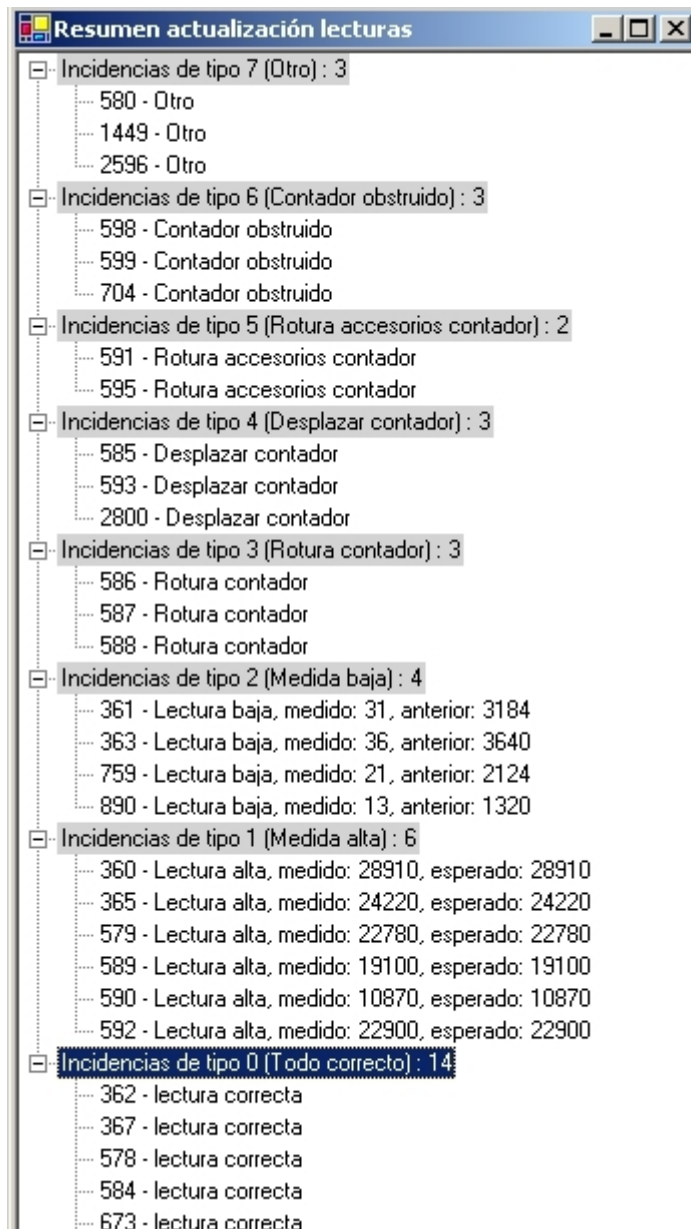
*Ilustración 43: Selección de la fecha de las lecturas e incidencias*

Si existen lecturas o incidencias que registrar, el proceso no continuará hasta que se precise la fecha con que deben ser registradas.

La tercera de estas interacciones permite al usuario estudiar el proceso de registro de las lecturas en un informe que resume los resultados de la descarga de lecturas e incidencias.

Este informe muestra en una vista de árbol las incidencias que se han dado, permitiendo expandir los nodos para estudiar detalladamente qué contadores han tenido las incidencias.





*Ilustración 44: Resumen de registro de lecturas e incidencias*

Durante todo el proceso de sincronización se presenta el estado y progreso en la pantalla principal de sincronización. Se informa de la correcta conexión y desconexión, de las operaciones iniciadas y terminadas, de los avances en las operaciones, de los errores que puedan ocurrir, y del tiempo empleado en todo el proceso.

```
#Actualizar orden-rutas iniciada#
• 0 ordenRutas recibidas
• OrdenRutas comparadas
• 0 ordenRutas a borrar enviadas
• 0 ordenRutas a actualizar enviadas
• 2867 ordenRutas a añadir enviadas
#Actualizar orden-rutas terminada#
#Actualizar configuracion iniciada#
#Actualizar configuracion terminada#
<Desconexión correcta>
#duración: 0m:59s.264ms#
```

*Ilustración 45: Progreso, desconexión y tiempo invertido*

Una vez terminado el proceso de sincronización, la pantalla principal de sincronización queda mostrando la información de todo el proceso. Es necesario cerrar esta pantalla para iniciar otro proceso de sincronización.

#### 4.1.2.- Impresión

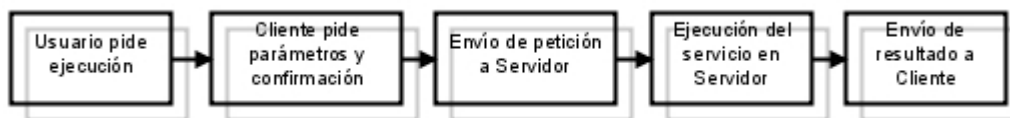
La funcionalidad de impresión de códigos de barras ha sido añadida en el modelo, de forma que automáticamente al generar el sistema mediante las herramientas de la metodología **OLIVANova** se crean los mecanismos necesarios que integran esta funcionalidad en la aplicación generada.

Ahora bien, la ejecución de servicios en un sistema generado por **OLIVANova** la realiza siempre la parte servidora, enviando los resultados al cliente. Esto supone un problema para nuestra funcionalidad de impresión de códigos de barras. La ejecución del servicio de impresión del código de barras en la parte servidora provocaría que la impresión se lanzase en el ordenador servidor, y que el código de barras acabara imprimiéndose en una impresora de la red a la que pertenece el servidor.

Queremos que la impresión se realice en la parte cliente, de forma que el usuario pueda escoger la impresora a emplear y tener acceso a los códigos de barras una vez impresos, pues conocemos la estructura del sistema y sabemos que las redes de cliente y servidor están separadas. Con tal de conseguir ejecutar el servicio en la parte cliente, **AguasCli**, manipularemos el código generado.

##### 4.1.2.a Manipulación del código generado automáticamente

Cómo se ha comentado, la ejecución de un servicio se realiza en el servidor. Al solicitar el usuario la ejecución de un servicio, se desencadena el mecanismo de invocación de servicios. Tras recoger del usuario los parámetros necesarios y al confirmar éste la ejecución del servicio, se envía la petición del servicio con todos los parámetros al servidor para su ejecución.



*Ilustración 46: Ejecución de servicios en OLIVANova*

Queremos evitar que se envíe un servicio al servidor, y ejecutarlo de forma local. Por esto manipularemos el código de forma que cuando el usuario haya introducido todos los parámetros y haya confirmado la ejecución se ejecute el servicio en la propia máquina en lugar de ser enviado al servidor.

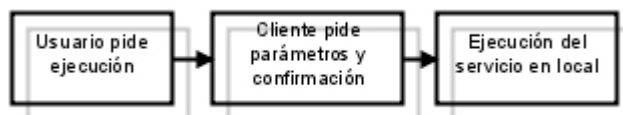
El lugar del código idóneo para evitar que se realice la petición al servidor y en su lugar se ejecute el servicio en local es el manejador del botón **Ok** del formulario que pide los parámetros y confirmación.



En ese punto, donde tenemos toda la información necesaria para la

ejecución del servicio y el usuario ha confirmado su invocación, modificaremos el código, eliminando la invocación al servidor y en su lugar colocaremos el código que se encarga invocar a la ejecución en local del servicio de impresión.

Así conseguimos que al invocar el servicio de impresión de códigos de barras desde la aplicación se ejecute de forma local tal servicio, ejecutándose toda la tarea en el ordenador corriendo **AguasCli**; con el resultado de imprimir los códigos de barras en la impresora deseada.

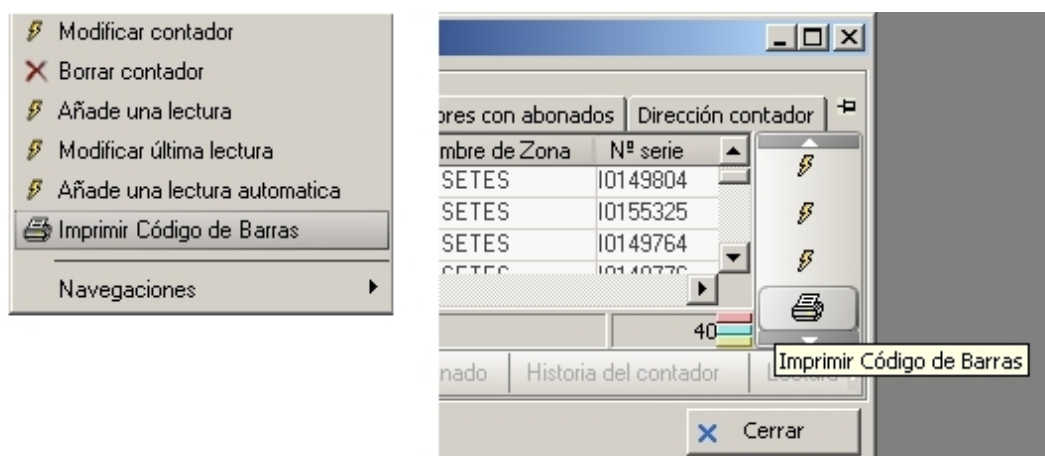


*Ilustración 47: Perversión de la ejecución del servicio*

#### **4.1.2.b Proceso de impresión**

El proceso de impresión del código de barras de un contador se ha integrado en la aplicación **AguasCli**, por lo que para imprimir un código se emplea un proceso idéntico al resto de servicios de la aplicación.

El servicio de impresión puede invocarse como cualquier otro servicio; ya sea a través del menú contextual o mediante la nueva operación de impresión que se añade en el formulario de contadores.

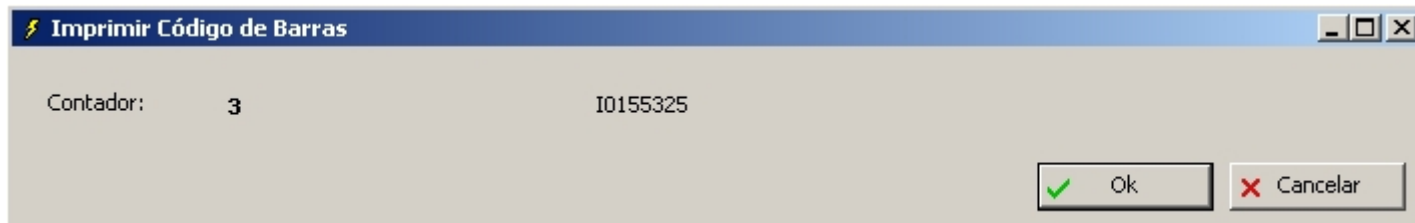


*Ilustración 48: Acceso a impresión de código de barras*

Una vez invocado el servicio, si no existía un contador seleccionado se pedirá identificar el contador para el que imprimir el código de barras :

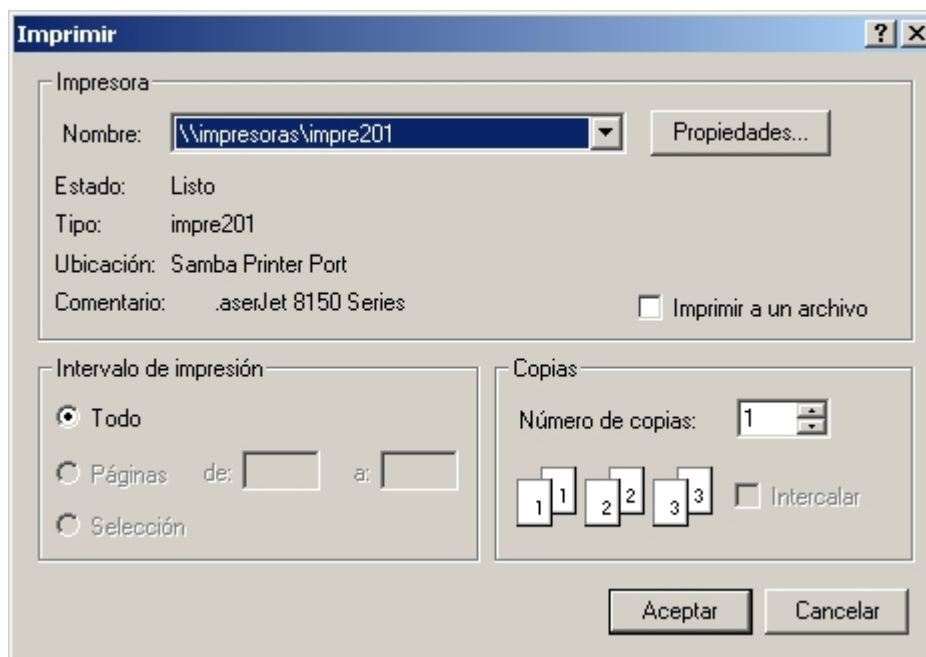


o se mostrará la información del contador ya seleccionado y se pedirá confirmación :



Al pulsar **Ok** se enviaría la petición de ejecución del servicio al servidor, pero en su lugar se ejecuta en local la impresión.

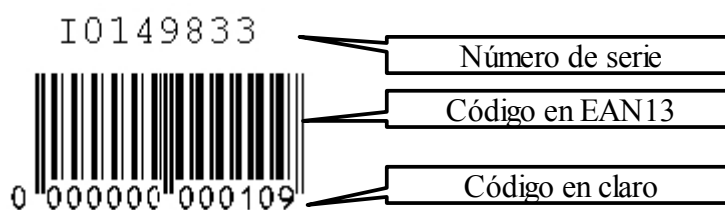
Para llevar a cabo la impresión, se siguen los pasos típicos de selección de impresora y configuración de las propiedades de impresión mediante un cuadro de diálogo de impresión:



*Ilustración 49: Diálogo de selección y configuración de impresora*

Una vez seleccionada la impresora y configurada la impresión, pulsando Aceptar se envía finalmente el documento a la impresora que terminará la impresión.

El resultado final es una etiqueta con el número de serie del contador y un código de barras que codifica el código del contador en EAN13.



*Ilustración 50: Código de barras obtenido*

## CAPÍTULO 5- Sincronización

En este capítulo se discute el proceso de sincronización diseñado entre la aplicación del dispositivo, **AguasCli** y el servidor. Se describe el protocolo diseñado, con la vista de cada una de las partes, las operaciones que se realizan, y el mecanismo de envío de datos; el papel de cada una de las partes en el proceso y los artefactos empleados para evitar la sobrecarga del servidor.

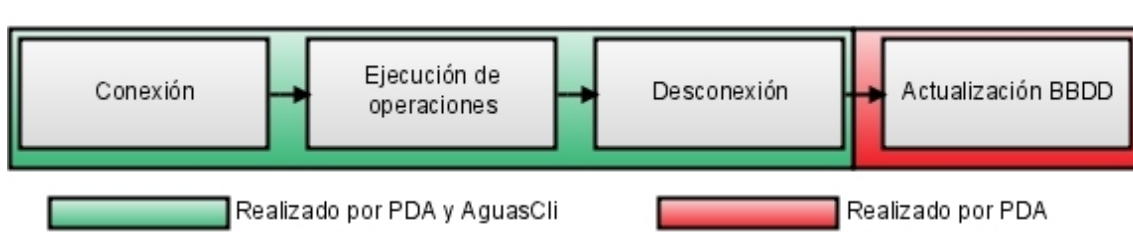
La sincronización, proceso esencial del sistema diseñado, permite mantener el dispositivo móvil al día, actualizando su base de datos con los últimos datos de que dispone el servidor, así como registrando en el servidor central las lecturas e incidencias pendientes de descarga. Es necesario disponer de conexión en el dispositivo para realizar la sincronización.

Durante la sincronización las tres partes del sistema tiene un papel importante. El dispositivo entrega nueva información (lecturas e incidencias) y recibe actualización de la información que contiene; la parte cliente, **AguasCli**, actúa de intermediario entre el dispositivo y el servidor, proporcionando una estructura ya construida para interactuar con el servidor central; y la parte servidora, **AguasSrv**, proporciona los últimos datos actualizados y guarda las lecturas e incidencias proporcionadas por el dispositivo.



*Ilustración 51: Movimiento de datos en sincronización*

El proceso se desarrolla en tres fases, más una fase en el dispositivo que no requiere conexión. El dispositivo inicia una conexión con **AguasCli** y, una vez conectados, es hora de llevar a cabo las operaciones que hayan sido establecidas para esta sincronización. Terminadas las operaciones, no queda más que desconectar el dispositivo y **AguasCli**. Tras la desconexión, el dispositivo actualiza su base de datos con los cambios enviados por **AguasCli**. Este último paso se realiza sin conexión y puede llevarlo a cabo el dispositivo autónomamente.



*Ilustración 52: Fases de sincronización*

Por diseño **AguasCli** tan sólo podrá atender las sincronizaciones de los dispositivos secuencialmente, de forma que si una petición de sincronización llega

mientras otra está en proceso tendrá que esperar a que acabe.

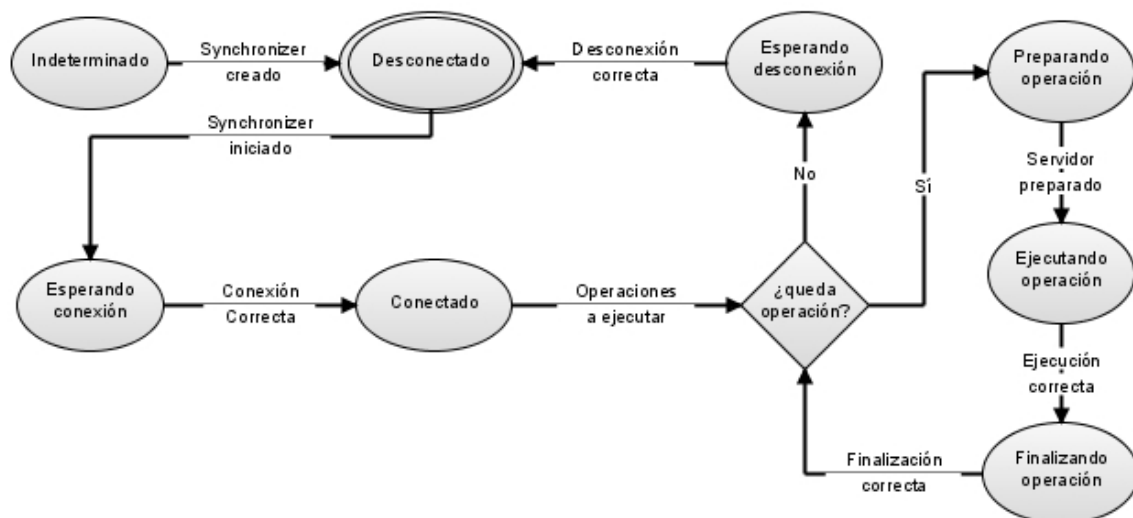
## 5.1.- Protocolo de sincronización

El proceso de sincronización de datos está dirigido y controlado por un protocolo específico para este propósito. El protocolo permite realizar la sincronización paso a paso, confirmando que ambas partes (dispositivo y [AguasCli](#)) están listas para avanzar a la siguiente fase. Este protocolo vigila el proceso mientras el dispositivo y [AguasCli](#) permanecen conectados. La conexión inicia el protocolo en ambas partes y la desconexión termina el protocolo en ambas partes.

El protocolo diseñado está basado en el intercambio de mensajes en ambos sentidos (message passing), enviando paquetes de señales para el avance del proceso y paquetes de datos para el intercambio de información. De esta forma cada parte conoce el estado de la otra parte y el estado del proceso.

### 5.1.1.- Protocolo en dispositivo

Así el protocolo en el dispositivo sigue este diagrama de estados con los posibles estados por que puede pasar el protocolo:



*Ilustración 53: Protocolo de sincronización en dispositivos*

El dispositivo es quien inicia el proceso de sincronización y toma el papel activo durante el proceso. Tras el establecimiento de conexión el siguiente paso es llevar a cabo las operaciones de sincronización. El dispositivo enviará peticiones al servidor hasta que éste le informe de que está preparado para servir la operación.



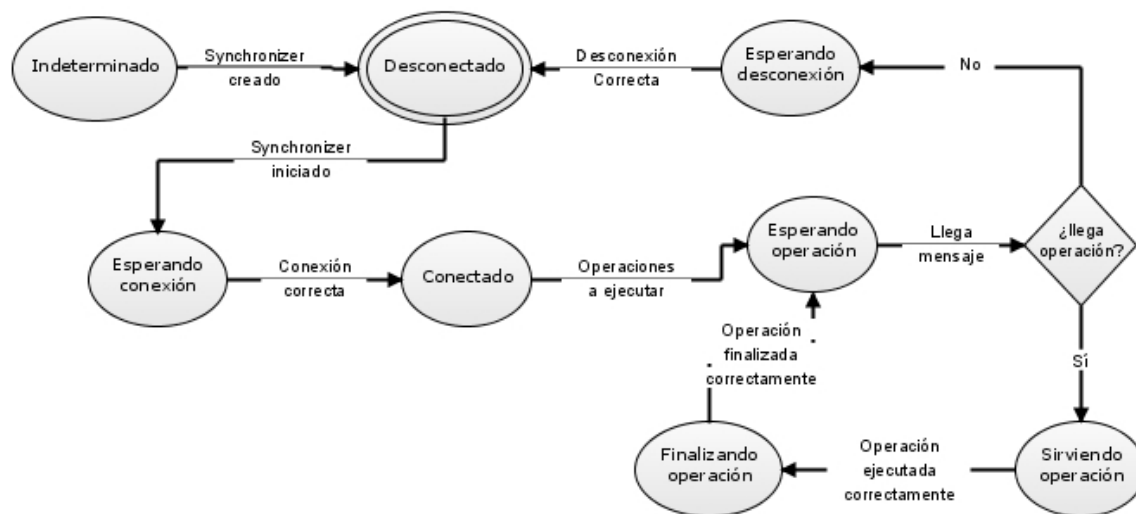
*Ilustración 54: Espera para ejecución de operación*

Las operaciones que pueden realizarse durante una sincronización son las encargadas de registrar las lecturas e incidencias en el servidor al tiempo que mantienen actualizados los datos del dispositivo. Las operaciones son : *Actualizar Lecturas, Actualizar Contadores, Actualizar Operarios, Actualizar Rutas, Actualizar OrdenRutas y Actualizar Configuración.*

Una vez ejecutadas las operaciones, se procede a la desconexión, terminando el protocolo de sincronización. En el dispositivo debe llevarse a cabo un paso más, la actualización de la base de datos para hacer persistentes los cambios. Este paso no requiere conexión, ya que se realiza localmente en el dispositivo y éste ya dispone de toda la información que necesita.

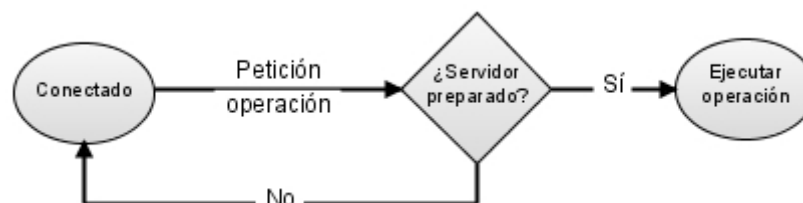
### 5.1.2.- Protocolo en **AguasCli**

El protocolo en **AguasCli** sigue este diagrama de estados con los posibles estados por que puede pasar el protocolo :



*Ilustración 55: Protocolo de sincronización en **AguasCli***

**AguasCli** es la parte servidora del protocolo, y como tal está a la espera de que un dispositivo se conecte y realice peticiones de operaciones. Si tras haber establecido conexión **AguasCli** se encuentra en una tarea que no lo permite atender operaciones puede retrasar la ejecución de las operaciones hasta que esté preparado.



Las operaciones que pueden servirse desde **AguasCli** son : *Actualizar Lecturas, Actualizar Contadores, Actualizar Operarios, Actualizar Rutas, Actualizar OrdenRutas y Actualizar Configuración.*

Una vez el dispositivo no tiene más operaciones a realizar, se procede a la desconexión y termina el protocolo.



### 5.1.3.- Operaciones del protocolo

El protocolo está diseñado con el propósito de dividir la actualización de los datos en operaciones aisladas que pueden ejecutarse por separado, de forma que se lleven a cabo únicamente aquellas operaciones que sean necesarias, minimizando el coste temporal de la sincronización.

Cada operación está encargada de actualizar unos datos, ya sea enviando información nueva al servidor central o ejecutando los cambios oportunos en el dispositivo. Las operaciones existentes son éstas : *Actualizar lecturas*, *Actualizar Contadores*, *Actualizar Operarios*, *Actualizar Rutas*, *Actualizar OrdenRutas* y *Actualizar Configuración*. De estas operaciones una de ellas, *Actualizar Lecturas*, registra nueva información en el servidor, mientras que el resto de operaciones mantienen al día los datos del dispositivo.

- **Actualizar Lecturas** : Esta operación recoge las lecturas e incidencias aún sin guardar del dispositivo para registrarlas en el servidor central, donde serán tratadas según la lógica de negocio definida ya en el sistema modelado en OLIVANOVA.

En esta operación el dispositivo recorre los contadores que tiene, y envía a **AguasCli** las lecturas e incidencias aún no registradas, borrándolas del dispositivo al terminar la operación.

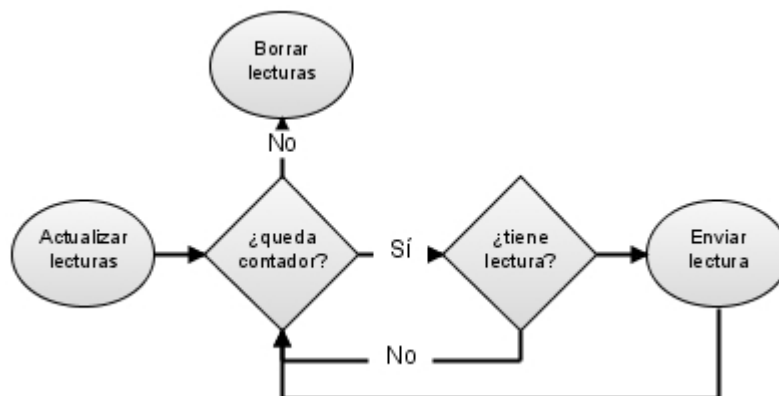


Ilustración 56: Operación actualizar lecturas en dispositivo

En esta operación **AguasCli** recibe las lecturas e incidencias enviadas por el dispositivo, registrándolas una a una en el servidor central, donde son tratadas según está establecido en el modelo en OLIVANOVA.



Ilustración 57: Operación actualizar lecturas en **AguasCli**

- **Actualizar [Contadores, Operarios, Rutas, OrdenRutas]** : Estas operaciones mantienen los datos del dispositivo actualizado. Los datos que manipulan estas operaciones son los datos con los que más trabaja el dispositivo en la realización de las tareas. Para estas cuatro tareas se sigue el mismo patrón de detección de cambios y actualización.

En estas operaciones el dispositivo envía los datos de que dispone, y espera a que **AguasCli** le envíe los cambios que debe realizar, en forma de lista de

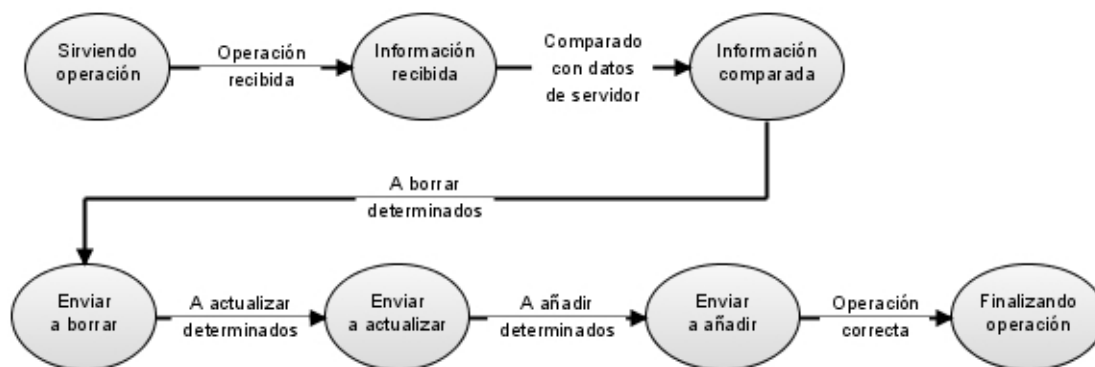


objetos a borrar, objetos a actualizar y objetos nuevos a añadir a la base de datos.



*Ilustración 58: Operación actualizar información en dispositivo*

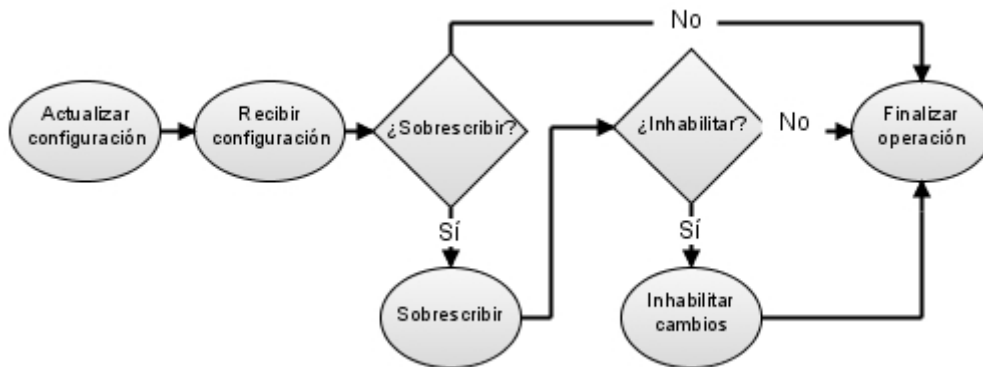
En estas operaciones **AguasCli** recibe los datos enviados desde el dispositivo, realiza un proceso de comparación entre los datos recibidos y los datos existentes en el servidor central. Del proceso de comparación **AguasCli** extrae los cambios en tres listas, una con los objetos a borrar, otra con los objetos que han cambiado y una última con los nuevos objetos a añadir. Estas listas son enviadas al dispositivo, que aplicará los cambios.



*Ilustración 59: Operación actualizar información en **AguasCli***

- **Actualizar Configuración** : Esta operación siempre se realiza durante una sincronización, y no es seleccionable por el usuario. Se encarga de, en caso que así esté especificado, sobrescribir la configuración del dispositivo con la indicada en **AguasCli**. Puede llegar a inhabilitar la posibilidad de cambiar la configuración en el dispositivo de forma local.

En esta operación el dispositivo espera la recepción de la configuración establecida en el servidor. Uno de los parámetros de la configuración indica si debe sobrescribirse la configuración del dispositivo con la del servidor. Si es así, se procederá a establecer los valores de los restantes parámetros a los indicados en el servidor. Otro de los parámetros indica si se permite cambiar la configuración del dispositivo de forma local. Si no se permitiera, se inhabilitarían los cambios en la configuración del dispositivo localmente, hasta que en otra sincronización se establezca que los cambios locales en el dispositivo son permitidos.



*Ilustración 60: Operación actualizar configuración en dispositivo*

En esta operación **AguasCli** busca entre las configuraciones de dispositivo disponibles aquella que pertenece al dispositivo que está sincronizándose y le envía al dispositivo los parámetros de esa configuración.

#### 5.1.4.- Discriminación de los cambios

Las operaciones *Actualizar Contadores*, *Actualizar Operarios*, *Actualizar Rutas*, *Actualizar OrdenRutas* y *Actualizar Configuración* tienen como propósito mantener al día los datos del dispositivo respecto de los datos del servidor.

En la ejecución de cada una de esas operaciones **AguasCli** recibe los datos de que dispone el dispositivo y los datos de que dispone el servidor. Una vez tiene los datos de ambas partes realiza una comparación entre los dos grupos de datos, discriminando los cambios a realizar que serán enviados al dispositivo.

El algoritmo de comparación recibe como entrada la lista de datos del dispositivo, tres listas a rellenar con los cambios (a borrar, a añadir, a actualizar), y un suministrador de datos del servidor.

El grueso del algoritmo es un bucle que va tomando uno a uno datos del servidor. En cada iteración el dato se compara con la lista de datos del dispositivo, dando como resultado que haya que:

- **Actualizarlo en el dispositivo** : si el mismo objeto se encuentra tanto en el dispositivo como en el servidor, pero tienen datos diferentes, será necesario actualizar el que se encuentra en el dispositivo con los datos del servidor.
- **Añadirlo al dispositivo** : si un objeto se encuentra en el servidor pero no en el dispositivo será necesario añadirlo al dispositivo con los datos del servidor.
- **Borrarlo en el dispositivo** : si un objeto se encuentra en el dispositivo pero no en el servidor, será necesario borrarlo del dispositivo.
- **No realizar ningún cambio** : si un objeto se encuentra en el dispositivo y en el servidor con los mismos datos, no será necesario realizar ningún cambio.

El pseudocódigo del algoritmo puede verse aquí:

```
Input : dataFromPDA, dataFromServerProvider
Output : dataToAdd, dataToDelete, dataToUpdate

do
    dSRV = dataFromServerProvider.Next
    if dSRV is NULL
        break
    end if
    dPDA = dataFromPDA[dSRV.Identifier]
    if dPDA is NULL
        dataToAdd.Add(dSRV)
    else
        if dSRV.Equals(dPDA) is FALSE
            dataToUpdate.Add(dSRV)
        end if
        dataFromPDA.Remove(dPDA.Identifier)
    end if
while TRUE

forall data d in dataFromPDA
    dataToDelete.Add(d)
end forall
```

Una vez discriminados los cambios y organizados en tres listas para borrar, actualizar y añadir, se envía al dispositivo cada una de esas listas haciendo efectivos los cambios.

### 5.1.5.- Envío de datos

Durante todo el proceso de sincronización el dispositivo y [AguasCli](#) intercambian mensajes y datos. Este intercambio se realiza a través de *streams* (flujos) de .Net, que normalmente se establecerán sobre sockets, pero no es obligatorio.

Estos *streams* facilitan el envío de datos evitando que tengamos que preocuparnos por detalles de compatibilidad a niveles bajos del enlace entre el dispositivo y [AguasCli](#).

Aprovechando la abstracción que nos ofrecen los *streams*, tanto los mensajes como los datos se envían en forma de cadena. Los mensajes siendo simples cadenas con información de confirmación o peticiones no requieren ningún tratamiento especial para su envío. En cambio los objetos a enviar deberán ser sometidos a una serialización que represente sus datos en forma de cadena.

Para la serialización de los objetos **.Net Framework** ofrece serializadores

XML para cualquier tipo de objeto, pero éstos resultan ser demasiado costosos computacionalmente, introduciendo importantes retardos en el proceso de sincronización.

Se optó entonces por implementar varios serializadores/deserializadores, bajo una interfaz común, específicos para cada tipo; buscando el mínimo coste computacional y espacial en la serialización de los objetos.

## **5.2.- AguasCli como intermediario**

En el proceso de sincronización el dispositivo puede tener datos que han de acabar en el servidor, mientras que el servidor puede tener datos que han de acabar en el dispositivo. Esto es, las partes del sistema que contienen datos manipulados durante la sincronización son el dispositivo y el servidor.

No obstante, **AguasCli** realiza un papel importante durante el proceso, como intermediario entre el dispositivo y el servidor, tomando parte de la carga que supone el proceso así como ayudando a que resulte menos complejo.

### **5.2.1.- Uso de la metodología OLIVANova**

Una de las principales razones para emplear a **AguasCli** como intermediario es la estructura basada en la metodología **OLIVANova** que tiene implementada. El acceso a los datos y servicios del servidor tal y como aparecen en el modelo puede realizarse únicamente mediante las librerías de acceso a datos y servicios proporcionadas por **OLIVANova**.

La estructura de una aplicación que acceda a los datos mediante esas librerías debe seguir la metodología de **OLIVANova**, realizando las inicializaciones adecuadas, empleando los sistemas de autenticación de agentes, utilizando los objetos apropiados para cada tipo de consulta y las peticiones de servicios por objeto.

La aplicación **AguasCli** que se genera automáticamente ya dispone de la estructura necesaria montada, por lo que si empleamos a **AguasCli** como intermediario podremos hacer uso de su estructura para nuestro propósito, sin perjudicar en el funcionamiento normal de la aplicación.

Es por esto que para llevar a cabo el proceso de sincronización se emplean los mecanismos de autenticación ya implementados en **AguasCli** con el fin de tener visibilidad y acceso a los datos y servicios que nos interesan. Empleamos también el sistema de configuración de acceso al servidor y de configuración global de la aplicación que se establece al arrancar **AguasCli**.

Al emplear a **AguasCli** como intermediario solucionamos uno de los problemas que planteaba la sincronización, la transferencia de datos entre el servidor y el dispositivo, de manera que el dispositivo se comunica con **AguasCli** con su propio protocolo conocido entre ellos pero no por el servidor, y **AguasCli** se comunica con el servidor empleando la metodología de **OLIVANova** que ambos conocen.

Veremos que utilizar un intermediario en la sincronización nos aporta también otras ventajas.

### 5.2.2.- Control de la carga del proceso

El proceso de sincronización, en especial la discriminación de los cambios en los datos, conlleva una carga de cómputo notable. Llevar a cabo esta tarea en el dispositivo no es viable debido a las limitaciones que nos impone la configuración hardware. El servidor puede encargarse fácilmente de esta tarea, pero es tal el volumen de datos a procesar que el servidor estaría varios segundos sobrecargado; algo que queremos evitar.

Encontramos aquí un nuevo papel que puede llevar a cabo **AguasCli** en su labor de intermediario. El equipo sobre el que se ejecuta **AguasCli** (típicamente un equipo de sobremesa) es capaz de soportar la tarea de discriminación de cambios, y el puesto de intermediario que toma dará pie a algunas optimizaciones.

De todas formas, aunque el servidor no se encargará de procesar la carga pesada del proceso, debemos establecer mecanismos que impidan saturar el servidor con peticiones de servicios o datos.

El mecanismo diseñado con el fin de controlar la carga a la que se somete al servidor por las peticiones de servicios y datos invocadas implica la restricción del número de peticiones que se realizan por intervalo de tiempo. Así se han definido dos parámetros, un intervalo de tiempo y un número de peticiones.

Se controla que en el intervalo de tiempo definido por el primer parámetro no se realicen más peticiones de las permitidas por el segundo parámetro. Manipulando ambos parámetros puede llegarse a la combinación ideal que no sature el servidor y permita llevar a cabo la sincronización en un tiempo aceptable.

La petición de datos del servidor se ha encapsulado en una clase, **InformationGetter**, que hace transparente el control de la carga.

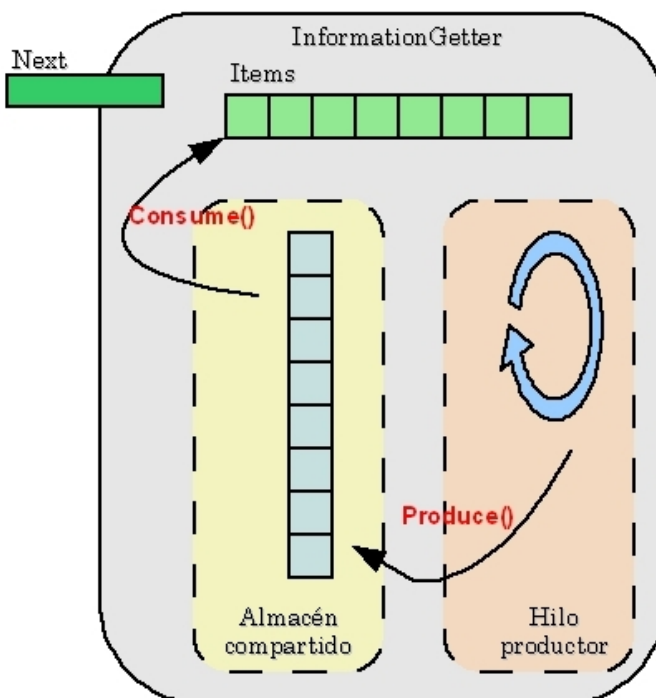
#### 5.2.2.a **InformationGetter**

Esta clase encapsula la petición de datos al servidor mientras mantiene un control de la carga a la que somete al servidor. Se encarga de proveer de información que contiene el servidor, almacenándola para que en posibles futuros usos no deba repetirse la petición de los mismos datos al servidor.

Emplea un almacén temporal donde un hilo productor va depositando los datos que solicita del servidor; y de ese almacén temporal los datos son trasladados y almacenados a otro almacén no compartido donde se guarda la información.

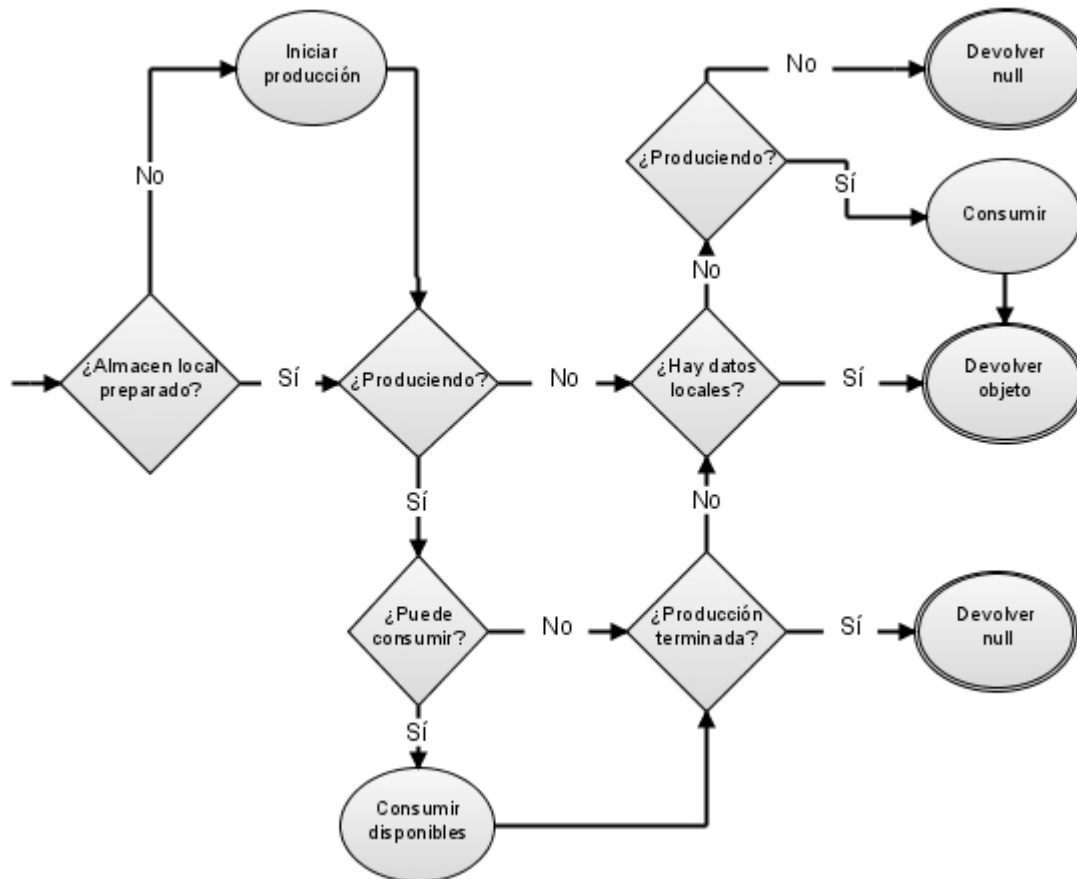
Se sigue el patrón clásico de productor-consumidor para ir rellenando el almacén compartido con los datos del servidor.

La información se ofrece a través de una propiedad que transparenta todo el proceso de producción y consumición del almacén compartido o de



*Ilustración 61: InformationGetter*

obtención de la información a nivel local. Al solicitar el siguiente elemento mediante esta propiedad se desencadena este algoritmo :



*Ilustración 62: Producción de datos*

Así esta clase puede emplearse para trabajar con información del servidor sin preocupaciones por tener que obtener esa información directamente del servidor ni de estar saturando el servidor al pedirle la información.

### **5.3.- Errores y cancelaciones**

Es obvio que durante el proceso de sincronización puede ocurrir algún error (en el canal de comunicación, en el acceso a la base de datos, etc.) que impida terminar correctamente el proceso. Así también puede que el usuario decida cancelar el proceso de sincronización antes de que éste termine. En ambos casos el proceso es interrumpido antes de haber completado todas las operaciones y debe decidirse un comportamiento a seguir ante estas situaciones.

#### **5.3.1.- Errores**

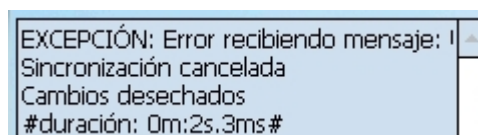
Al ocurrir un error durante el proceso de sincronización ninguna de las partes puede asegurar el estado en que se encuentra la otra parte. Si estaba ejecutándose una operación no hay forma de saber si esa operación llegó o llegará a terminar correctamente.

De la correcta información de los datos que contiene el dispositivo depende gran parte de la corrección de las lecturas que se tomen durante el el proceso de

toma de lecturas de los contadores. No es permisible pues que haya datos incorrectos o incoherentes en el dispositivo a raíz de errores en la sincronización.

Es por esto que ante la aparición de errores durante la sincronización se toman estas medidas:

- **Cancelación de la sincronización** : el proceso de sincronización se cancela tanto en el dispositivo como en [AguasCli](#), y tendrá que iniciarse de nuevo para llevarlo a cabo correctamente.
- **Descarte de los cambios ya recibidos** : si se hubieren recibido cambios a aplicar en el dispositivo antes de la aparición del error, éstos serán descartados y no se aplicarán en el dispositivo.
- **Cierre del canal de comunicación** : el canal de comunicación, esto es, los *streams* empleados para llevar a cabo la comunicación entre el dispositivo y [AguasCli](#) son cerrados en ambas partes.
- **Mostrar mensaje de error en el log** : en el área de log de la pantalla de sincronización se muestra un mensaje con el error que ha ocurrido y que ha provocado la cancelación del proceso.



*Ilustración 63: Error de sincronización en dispositivo*

Tras un error, los datos del dispositivo no han cambiado y puede retomarse sin problemas un nuevo proceso de sincronización.

### 5.3.2.- Cancelaciones

La cancelación de la sincronización puede hacerse desde el la aplicación del dispositivo o desde [AguasCli](#), indistintamente. Desde el dispositivo la sincronización puede cancelarse pulsando el botón Cancelar de la pantalla de [sincronización](#). Desde [AguasCli](#) el proceso puede cancelar pulsando el botón Cancelar de la ventana de [sincronización](#). En ambos casos se pedirá confirmación para cancelar el proceso.



*Ilustración 64: Botones para cancelar sincronización*

La cancelación del proceso de sincronización tiene mucha similitud con la aparición de un error, tan sólo que esta situación permite hacer una parada controlada del proceso. En este caso se toman las mismas medidas que ante la aparición de un error, tan sólo que bajo condiciones controladas y con conocimiento de ambas partes.

- **Cancelación de la sincronización** : se detienen los hilos del proceso de sincronización, deteniendo el proceso en sí en ambas partes.
- **Descarte de los cambios recibidos** : los cambios que haya recibido el dispositivo se descartan, puesto que la aplicación de los cambios es parte del proceso de sincronización, y al haber cancelado el proceso se cancelan

también los cambios.

- **Cierre del canal de comunicación** : el dispositivo y [AguasCli](#) cierran ordenadamente sus canales de comunicación.
- **Mostrar mensaje de cancelación en el log** : se muestra en el área de log un mensaje informando de la cancelación de la sincronización.

Tras una cancelación, los datos del dispositivo no han cambiado y puede retomarse sin problemas un nuevo proceso de sincronización.



## CAPÍTULO 6- Códigos de barras

En este capítulo se discute la inclusión de códigos de barras en el proceso de toma de lecturas con el propósito de una rápida y libre de errores identificación de los contadores. Son expuestos los más utilizados estándares, seleccionando uno de ellos. Se especifican en detalle las características técnicas del escáner del dispositivo **Datalogic Blackjet™**, dispositivo sobre el que se ha diseñado el trabajo de campo. Por último se describe en profundidad la integración de la utilización de los códigos de barras en todo el sistema.

Emplear códigos de barras proporciona una solución a la necesidad de una rápida, fiable y libre de errores identificación de los contadores de agua. El formato dado a las etiquetas deberá también permitir identificar un contador aún sin disponer de un lector de códigos de barras.

### 6.1.- Lector de códigos de barras del dispositivo

El dispositivo **Datalogic Blackjet™** cuenta con un lector láser de códigos de barras lineales. Éstas son sus especificaciones técnicas :

Especificaciones ópticas	
Resolución máxima	0,13 mm / 5 mils
Inclinación lateral	± 55°
Inclinación vertical	± 65°
Ratio de escaneo	35 ± 5 escaneos por segundo
Fuente de luz	VLD, longitud de onda 630~680 nm
Seguridad	Class II EN 60825-1/CDHR

*Tabla 4: Especificaciones del lector de códigos de barras*

Puede decodificar estos estándares : UPCA A, UPC E, EAN 8, EAN 13, Código 39, Código 39 ASCII, Entrelazado 2 de 5, Industrial 2 de 5, Matriz 2 de 5, Código 128, Codabar, MSI, Plessey, Código 93 y Código 11.

El láser del lector tiene capacidad de leer códigos de barras hasta a más de 70 cm de distancia, aunque a mayor distancia menor es la resolución que presenta el láser. Este diagrama muestra la resolución del láser en función de la distancia, comenzando con 0,13 mm de resolución a unos 10 cm de distancia y llegando a 1,00 mm de resolución a más de 70 cm de distancia.

### 6.2.- Códigos de barras existentes

Los códigos de barras emplean una representación de la información con objetivo de resultar fácil para una máquina; procurando codificar la máxima información en el menor espacio. [12]

### 6.2.1.- Formato de un código de barras

El código de barras está formado por barras verticales de diferente grosor separadas por espacios en blanco. El grosor de cada uno de esas barras o espacios en blanco viene dado en módulos (unidad de medida específica de los códigos de barras) o en mils (milésimas de pulgada). Así dependiendo de la simbología empleada al dar formato al código el grosor de las barras o los espacios puede tener diferente significado. Las simbologías pueden clasificarse en dos grupos atendiendo a dos criterios diferentes :

- *Continuo frente a discreto* : los caracteres en las simbologías continuas, comienzan con un espacio y el siguiente comienza con una barra, o viceversa. Los caracteres en las simbologías discretas comienzan y terminan con barras; el espacio entre caracteres es ignorado, en cuanto no es lo suficientemente ancho para parecerse al final del código.
- *Bidimensional frente a multidimensional* : las barras en las simbologías bidimensionales son anchas o estrechas; cuanto sean de anchas, no importa y pueden variar de un carácter al siguiente. Las barras en las simbologías multidimensionales son múltiplos del módulo del código de barras; generalmente, se emplean barras con anchura 1, 2, 3 ó 4 módulos.

### 6.2.2.- Clasificaciones de códigos de barras

Los diferentes tipos de códigos pueden clasificarse según dos cualidades, el dominio que pueden codificar y como realizan la codificación. Así tenemos que los códigos de barras pueden categorizarse de esta forma:

- Por el dominio que puede representar :
  - Sólo dígitos : este tipo de códigos de barras representan únicamente los dígitos del 0 al 9. Puede variar la longitud del código y el tipo de codificación empleada y normalmente incluyen un dígito de control. Un ejemplo:



*Ilustración 65: Código de barras numérico*

- Caracteres y dígitos : este tipo de código de barras representan un dominio mayor, incluyendo un conjunto de caracteres alfanuméricos entre los valores representables. Un ejemplo :



*Ilustración 66: Código de barras alfanumérico*

- Por la representación que emplea :
  - Lineal : la información está codificada de forma que se requiere la lectura de una sola dimensión para decodificarla, esto es, un línea de la imagen que representa el código de barras basta para interpretarlo. Para mejorar la legibilidad en condiciones complicadas pueden tomar una forma circular que asegura su legibilidad desde todas las direcciones. Un ejemplo :



*Ilustración 67: Códigos de barras lineales*

- 2D : la información está codificada en una imagen, normalmente cuadrada, de forma que es necesario leer dos dimensiones para decodificar la información. Son menos empleados, pero pueden guardar más información en el mismo espacio. Un ejemplo :



*Ilustración 68: Código de barras 2D*

### 6.2.3.- Estándares más utilizados

Con diferencia, el tipo de código de barras más empleados son los lineales, y entre ellos se destacan estos estándares:

- **Código 39** : permite codificar los dígitos de 0 a 9 y caracteres en mayúsculas de la A a la Z. Emplea una longitud fija para cada carácter a representar, por lo que la longitud total es variable según la cantidad de información que incluya el código. Puede realizarse una traducción directa desde el código en claro mediante fuentes TrueType.



*Ilustración 69: Formato Código 39*

- **Código 128** : permite codificar la tabla ASCII entera, con sus 128 caracteres. Con técnicas de alta densidad consigue codificar dos caracteres en el espacio de uno. La longitud del código dependerá de la cantidad de

información a codificar. Realizar la conversión de código en claro a código de barras requiere el empleo de herramientas profesionales.



*Ilustración 70: Formato Código 128*

- **Entrelazado 2 de 5** : permite codificar únicamente dígitos. Emplea el espacio de un dígito para representar dos dígitos aprovechando las barras para codificar uno de ellos y los espacios para representar el otro. La longitud del código dependerá de la cantidad de dígitos (que tendrá que ser par) que quiera codificarse. Realizar la conversión del código en claro a código de barras requiere el empleo de herramientas profesionales.



*Ilustración 71: Formato Entrelazado 2 de 5*

- **UPC (Universal Product Code)** : estándar ampliamente difundido por EEUU y Canadá; permite codificar únicamente 12 dígitos, resultando en una longitud fija. Emplea segmentos de 7 bits (o módulos) para representar cada dígito, separándolos mediante barras de guarda que delimitan el inicio, final y parte media del código. Realizar la conversión del código en claro a códigos de barras requiere el empleo de herramientas profesionales.



*Ilustración 72: Formato UPC*

- **EAN (European Article Number)** : estándar ampliamente difundido por Europa y Japón; permite codificar únicamente dígitos, con longitudes prefijadas de 8, 13, 14 ó 17 dígitos, más un dígito de control. El código está dividido en tres campos que identifican el país, el fabricante y el producto. Realizar la conversión del código en claro a código de barras requiere el empleo de herramientas profesionales.



*Ilustración 73: Formato EAN13*

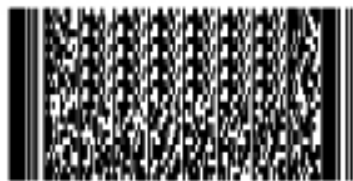
Dentro de los códigos de barras 2D los estándares más empleados son :

- **Datamatrix** : permite codificar texto o datos binarios, entre 55 y 2K bits de información. Los datos se codifican en una matriz de entre 8x8 a 144x144 bits codificados con un blanco (0) o un negro (1). El código de barras final puede ser tan pequeño como 2 milímetros cuadrados. Realizar la conversión de código en claro a código de barras requiere el empleo de herramientas profesionales.



*Ilustración 74: Formato DataMatrix*

- **PDF417** : permite codificar texto o datos binarios, hasta 2710 caracteres. Representa la información en líneas codificando los bits con blancos o negros. Un código de barras puede tener de 3 a 90 líneas, todas ellas delimitadas por dos zonas que marcan el inicio y el final de la línea. Realizar la conversión del código en claro a código de barras requiere el empleo de herramientas profesionales.



*Ilustración 75: Formato PDF147*

### **6.3.- Código de barras elegido**

Al escoger el tipo de código de barras que utilizaremos en nuestro sistema tendremos en cuenta estos factores :

- **Representaremos dígitos** : los contadores se identificarán mediante su número de contador, un campo entero representable mediante dígitos.
- **Tamaño de código fijo** : el código de barras se imprimirá sobre etiquetas de tamaño fijo, por ende, el código de barras no podrá sobrepasar cierto tamaño.
- **Legible por el lector del dispositivo** : debe emplearse un estándar que el lector de códigos de barras del dispositivo pueda decodificar.
- **Permita crecer el número de contadores** : el volumen de contadores a medir crecerá con el tiempo, de forma que el estándar empleado debe permitir, o bien representar un número grande o no poner límite.
- **Facilidad de conversión** : la conversión de código en claro a código de barras deberá ser sencilla, procurando evitar el uso de herramientas de pago.

Enfrentando los tipos de códigos de barras existentes con los factores que marcarán qué estándar escogeremos podemos eliminar aquellos que no se ajustan a nuestras necesidades.

Para comenzar podemos descartar los códigos de barras de 2 dimensiones puesto que el lector de códigos de barras del dispositivo lee tan sólo códigos de barras lineales. Rechazamos pues los estándares **Datamatrix** y **PDF417**.

El tamaño permitido para los códigos de barras, limitado por el soporte físico sobre el que se imprimirán, nos aconseja emplear un estándar que tenga un tamaño fijo para el código de barras. Esta decisión nos hace descartar los estándares **Código 39**, **Código 128** y **Entrelazado 2 de 5**; quedando los estándares **UPC** y **EAN**.

Ambos estándares, **UPC** [13] y **EAN** [14], pueden ser decodificados por el lector de códigos de barras del dispositivo, y con la longitud de código que emplean pueden representar 1 billón de códigos diferentes, margen más que suficiente para soportar el crecimiento de contadores.

El factor que marcará entonces que estándar será empleado será la facilidad de conversión de un código en claro al código de barras. Las herramientas profesionales convierten cadenas con el código en claro a códigos de barras con el tamaño indicado; y muchas de estas herramientas proporcionan librerías de desarrollo para integrar la generación de códigos de barras en aplicaciones propias.

Ahora bien, todas las herramientas de generación de códigos de barras eran de pago, o versiones *shareware* con limitaciones en su uso. No obstante se encontró un algoritmo público [15] que, junto a una fuente TrueType (CodeEAN13, ean13.ttf) especialmente diseñada para ello, permite la conversión de un código en claro a **EAN13**. Aplicando este algoritmo puede generarse una cadena a partir del código original que, representada con la fuente diseñada para ello, resulta en un código de barras **EAN13**.

Puesto que ni **UPC** ni **EAN** ofrecen ninguna ventaja sobre el otro estándar, la posibilidad de convertir códigos a formato **EAN13** de manera gratuita inclinará la balanza a favor del estándar **EAN13**, que será el formato a emplear en los códigos de barras.

## 6.4.- Impresión

La integración de un mecanismo de identificación de los contadores mediante códigos de barras no estará completa sin la posibilidad de extraer el código de un contador en un código de barras con que identificarlo.

La manera más sencilla de obtener un código de barras para un contador en la impresión, que a raíz del entorno en el que se ubicarán se hará sobre un material plastificado resistente.

Desde el principio se intenta integrar completamente la impresión en la aplicación ya existente, con la facilidad de uso y flexibilidad en el soporte de impresión siempre en mente.

### 6.4.1.- Nueva funcionalidad

Con tal de conseguir integrar al máximo la impresión de códigos de barras para los contadores al tiempo que se facilita el uso, se añade una nueva funcionalidad en la aplicación **AguasCli** ya existente. Es proceso de integración del código y de impresión de un código de barras está detallado en el [capítulo 4](#).

El desarrollo de la nueva funcionalidad abarca tres apartados diferentes, cada uno encargado de una tarea específica de la impresión.

- **Funcionalidad nueva en el modelo** : se ha añadido al modelo en **OLIVANova** una nueva funcionalidad de impresión de códigos de barras. En el modelo esta funcionalidad está vacía y será ocupada por una funcionalidad implementada a mano que realice la tarea.
- **Codificación de los códigos** : se ha implementado el algoritmo descrito en <http://grandzebu.net/index.php?page=/informatique/codbar/ean13.htm>, que junto con la fuente TrueType indicada permite imprimir el código de barras.
- **Impresión** : se ha implementado el proceso de impresión, ajustando el tamaño del código de barras a imprimir al tamaño de la etiqueta que se colocará en los contadores. En el proceso de impresión se añade encima del contador el número de serie del contador como identificador adicional alternativo.

La agrupación de estas tres tareas permite imprimir un código de barras a partir de un contador desde la aplicación ya existente.

#### 6.4.2.- Impresora

La nueva funcionalidad de impresión de códigos de barras permite imprimir en cualquier impresora instalada en el equipo sobre el que se ejecuta **AguasCli**. Ahora bien, con tal de obtener códigos de barras impresos sobre un soporte que pueda resistir en ambientes de humedad y a la intemperie emplearemos una impresora de transferencia térmica sobre etiquetas plastificadas.

La impresora a emplear y para la que está especialmente diseñado el proceso de impresión es un modelo **Zebra S4M ZPL 200dpi**. [16].



*Ilustración 76: Impresora Zebra S4M ZPL 200 dpi*

Sus especificaciones técnicas son:

<b>Comunicación</b>	
Interfaz	Puerto serie RS-232 USB 1.1 Puerto paralelo de 8 bits IEEE 1284
<b>Impresión</b>	
Resolución	230 puntos/pulgada   8 puntos / mm
Tamaño del punto	0,00492 pulgadas   0,125 mm
Anchura máxima impresión	4,09 pulgadas   104 mm
Dimensión módulo código barras	5 mils a 50 mils
Velocidades de impresión por segundo	2 pulg   51 mm 3 pulg   76 mm 4 pulg   102 mm 5 pulg   127 mm 6 pulg   152 mm
<b>Papel de impresión</b>	
Longitud mínima	0,7 pulg   17,8 mm
Longitud máxima	39 pulg   991 mm
Anchura mínima	0,75 pulg   19 mm
Anchura máxima	4,5 pulg   112 mm
Espesor mínimo	0,003 pulg   0,076 mm
Espesor máximo	0,010 pulg   0,25 mm
Contraste marca negra en ODU	> 1,0 ODU
Contraste máximo del papel	<= 0,5 ODU
<b>Cinta de impresión térmica</b>	
Anchura mínima	2 pulg   51 mm
Anchura máxima	4,3 pulg   109 m
Longitud estándar proporción 2:1	984 pies   300 m
Longitud estándar proporción 3:1	1476 pies   450 m
Diámetro interno rollo cinta	1 pulg   25,4 mm

*Tabla 5: Especificaciones de impresora Zebra S4M ZPL 200 dpi*

### 6.4.3.- Soporte físico

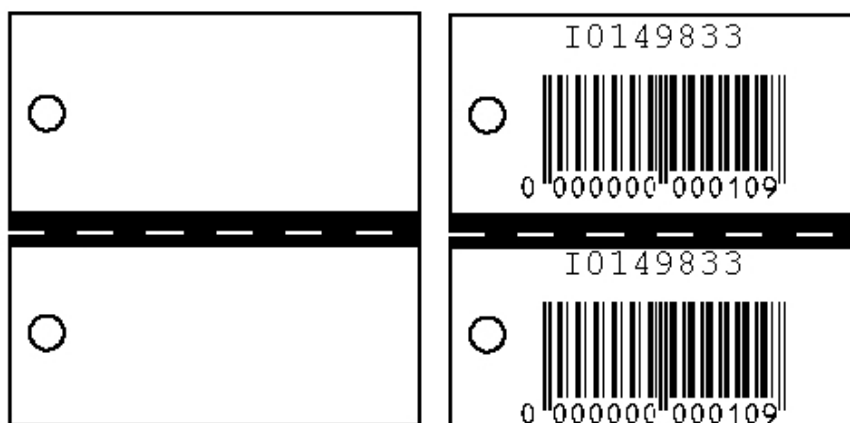
Los ambientes en los que se encontrarán los códigos de barras, con altos niveles de humedad y/o a la intemperie, nos obliga a emplear en la impresión un material apto para soportar tales condiciones.

Las etiquetas sobre las que se imprimirán los códigos de barras han sido fabricadas con **Valéron®** [17], un material que, aunque muy fino, es resistente a la humedad, difícil de rasgar o perforar, altamente resistente a la intemperie y



retardador del fuego. Puede comprobarse que este material se ajusta a nuestras necesidades.

Las etiquetas empleadas tendrán este formato :



*Ilustración 77: Formato de etiqueta y código de barras*

#### **6.4.4.- Impresión de todas las rutas**

La funcionalidad añadida a [AguasCli](#) permite imprimir un código de barras para un contador. Ahora bien, hay en funcionamiento cerca de 3000 contadores a la hora de integrar la identificación mediante códigos de barras en el sistema. Llevar a cabo la impresión de los códigos de barras de los contadores ya existentes mediante este método no es viable, con lo que se diseñó una solución alternativa.

Un factor a tener en cuenta antes de proceder a la impresión de los cerca de 3000 contadores es el orden que se les dará a los contadores impresos. Debe escogerse un criterio de ordenación que después permita una fácil colocación de las etiquetas en los contadores. Es aquí donde las rutas juegan un papel importante. Siguiendo el orden ya establecido en las rutas podemos imprimir los códigos de barras de manera que resulte sencillo localizar que etiqueta corresponde a cada contador.

Con tal de llevar cabo la impresión de los contadores ya en funcionamiento se optó por implementar una funcionalidad extra en [AguasCli](#) que no existirá en la versión final de [AguasCli](#). La nueva parte implementada permite seleccionar una de las rutas existentes, una impresora y procederá a imprimir la ruta con los contadores ordenados en la impresora seleccionada. Así podemos imprimir rápidamente las etiquetas con los códigos de barras ya ordenadas y preparadas para su colocación.



## CAPÍTULO 7- Implantación

Una vez conocido el proceso de toma de lecturas, analizado, diseñada una solución y desarrollada solo queda el paso final de implantación. La implantación estuvo marcada por dos etapas, de validación y de implantación.

Antes de llevar a cabo la implantación definitiva se realizó una visita a la empresa **Aguas del Bullent S.A.** para obtener la validación de la aplicación por parte de los usuarios finales.

En la implantación final se interactuó con **INTEGRANOVA**, [18], empresa encargada del mantenimiento de la aplicación, que realizó la implantación de la nueva versión de la aplicación, actualizando las partes servidora y cliente.

### 7.1.- Primera etapa, sugerencias, cambios, mejoras

La primera etapa de la implantación consistió en una visita a **Aguas del Bullent S.A.** en la que se mantuvo una entrevista con el personal de la empresa. El objetivo de esta visita era presentar el sistema desarrollado en una versión muy próxima a la final para que hubiera un primer contacto por parte de los operarios que iban a emplearlo.

De esta primera presentación del sistema se recogería información en forma de sugerencias y cambios a realizar sobre el sistema, tanto en la aplicación de los dispositivos (**waterLU**) como en la aplicación de escritorio (**AguasCli**). Una vez desarrollados los cambios y mejoras se procedió a poner en funcionamiento el sistema en una implantación final, coordinada con **INTEGRANOVA**.

#### 7.1.1.- Visita a Aguas del Bullent S.A.

Durante la primera visita a **Aguas del Bullent S.A.** se mantuvo una entrevista con los usuarios finales del sistema. El estadio de desarrollo que presentaba el sistema permitió presentar todas las nuevas funcionalidades, de forma que se evaluó el sistema al completo.

El personal presente en esta entrevista de validación fue :

- **Personal técnico** : operarios encargados de la instalación de contadores, el mantenimiento y la toma de lecturas. Usuarios finales de los dispositivos y de la aplicación **waterLU** desarrollada para la toma de lecturas. Evaluaron la usabilidad de la aplicación **waterLU** respecto a sus hábitos en el proceso de toma de lecturas.
- **Personal de oficina** : encargados de la gestión de los contadores, de la facturación y de las altas y bajas de contadores. Usuarios finales de la aplicación de escritorio con las nuevas funcionalidades **AguasCli**, así como de parte de la aplicación de los dispositivos **waterLU** que realiza la sincronización. Evaluaron la integración del nuevo sistema con el existente así como la funcionalidad que ofrecía la nueva funcionalidad respecto a la requerida.

En el proceso de entrevista se tomaron notas sobre los cambios y mejoras a aplicar antes de llevar a cabo la implantación.

### 7.1.2.- Personal técnico

Encargados del trabajo de campo, instalando los nuevos contadores, realizando el mantenimiento y llevando a cabo la toma de lecturas, serán los usuarios de los nuevos dispositivos y por ende los principales evaluadores tanto de los dispositivos como de la aplicación **waterLU** desarrollada para dar soporte al proceso de toma de lecturas.

Durante la entrevista y evaluación de la aplicación de dispositivo **waterLU** la atención se centró en la usabilidad de la aplicación, aunque no fue el único punto destacado :

- **Facilidad en la identificación de contadores** : uno de los mayores inconvenientes del sistema antiguo era la correcta identificación de un contador antes de su lectura. En este punto la nueva aplicación **waterLU** ofrecía una identificación automatizada por códigos de barras o seguimiento de rutas, así como una introducción manual de códigos de contadores que resultó satisfacer los requisitos de una identificación rápida y correcta.
- **Facilidad en la toma de lecturas** : la toma de lecturas pretendía ser lo más automática posible y asistir en la validación del valor introducido. Para ello la aplicación ofrecía una predicción del valor de consumo, así como botones de incremento y decremento variable según el histórico de consumo para realizar ajustes a ese valor, y un control de valores anómalos.

Estas ayudas fueron bien recibidas, ahora bien, en la información mostrada durante la toma de lecturas debía aparecer el valor de la última lectura de forma que el operario pudiera verificar el valor visualmente y dejar que la máquina lo confirme.

- **Navegación en rutas** : en el sistema antiguo los contadores están organizados por orden en diferentes rutas. Esta organización debía mantenerse en el nuevo sistema, y permitir una navegación y seguimiento de rutas flexible. La aplicación permite la selección de una ruta a seguir, tomando lectura de los contadores de esa ruta según su orden dictamina, y en cualquier momento puede abandonarse la ruta para tomar lectura de un contador fuera de la ruta.

Esta parte del sistema era satisfactoria por completo excepto en el punto de toma de lecturas fuera de la ruta seguida. Al tomar lectura de un contador fuera de la ruta seguida debía seguirse la ruta en el punto en que se abandonó.

- **Interfaz** : tanto por el perfil de los usuarios finales como por las condiciones del entorno de trabajo, la interfaz de la aplicación **waterLU** debía ofrecer una interacción libre de la necesidad de precisión y de condiciones lumínicas óptimas. Por ello el sistema ofrecía elementos de interacción diseñados con tamaño grande, amplia separación entre ellos, caracteres claros y contrastados y baja densidad de elementos de interacción.

Esta parte del sistema no recibió críticas tras demostrarse la posibilidad de llevar a cabo todas las operaciones con el uso de la pantalla táctil con los dedos, sin emplear elementos de precisión como la pluma.

- **Resistencia de los dispositivos a los golpes y al agua** : por el entorno de trabajo en el que van a emplearse los dispositivos, es crucial una buena resistencia a los golpes y al agua. El estándar IP64 con que cumplen los dispositivos garantizan un buen funcionamiento en el lugar de trabajo.
- **Duración de la batería** : los dispositivos serán empleados intensivamente

durante las jornadas de toma de lecturas de contadores. Se requiere una duración de aproximadamente 8 horas con el dispositivo ejecutando la aplicación y empleando el lector de códigos de barras del dispositivo. Ante este requerimiento se tomó la decisión de emplear baterías de larga duración que pueden proporcionar aproximadamente el doble de tiempo de carga al dispositivo.

Los cambios sugeridos y los requerimientos aún no desarrollados fueron anotados para introducirlos en la aplicación antes de la implantación final.

### 7.1.3.- Personal de oficina

Encargados de llevar la gestión de **Aguas del Bullent S.A.**, su principal herramienta de trabajo es la parte cliente **AguasCli** de la aplicación generada con **OLIVANOVA**. Serán los usuarios finales de las nuevas funcionalidades integradas en **AguasCli** así como de la parte de sincronización de los dispositivos desde **waterLU**.

Durante la entrevista y evaluación de los añadidos a **AguasCli** la atención se centró en la usabilidad y en la información mostrada en :

- **Facilidad de sincronización** : los dispositivos sincronizan sus datos con los del servidor central a través de **AguasCli**, y este proceso de sincronización ha de ser lo más automático posible. La única interacción requerida para llevar a cabo una sincronización es la que inicia el proceso; los pasos siguientes se realizan automáticamente.

La virtualmente total automatización del proceso de sincronización fue acogida sin críticas.

- **Registro de lecturas e incidencias** : durante el proceso de toma de lecturas de contadores la aplicación del dispositivo recoge lecturas e incidencias que se registrarán el servidor central al llevar a cabo una sincronización. Este proceso está automatizado dentro de la sincronización. Ahora bien, antes de llevar a cabo el registro de las lecturas e incidencias en el servidor central, el usuario ha de introducir la fecha a la que corresponden esas lecturas e incidencias.
- **Impresión de códigos de barras** : los contadores que se den de alta desde la implantación de este nuevo sistema necesitarán un código de barras impreso sobre etiquetas plastificadas para instalarlo junto con el contador. Una impresora térmica y una nueva funcionalidad en **AguasCli** se encargarán de facilitar esta tarea.
- **Integración en el sistema** : la integración de todas las nuevas funcionalidades en el sistema ya existente, sin el empleo de herramientas o aplicaciones nuevas o fuera de las ya empleadas permite continuar con la misma dinámica de trabajo, así como proporciona una buena base para una futura evolución del sistema.

Los cambios sugeridos y los requerimientos aún no desarrollados fueron anotados para introducirlos en el sistema antes de la implantación final.

Tras la entrevista y primera presentación del sistema, y habiendo recogido y anotado las sugerencias y cambios a realizar se procedió a analizar esta información recogida directamente de los usuarios finales.

Se inició un pequeño proceso de análisis, diseño y desarrollo de los nuevos requerimientos recogidos. Al terminar con este desarrollo se procedió a implementar finalmente el sistema.

## 7.2.- Segunda etapa, implantación en un escenario real.

La fase última de la implantación integró finalmente el nuevo sistema de toma de lecturas en el sistema anterior. Tras haber sido actualizada la aplicación a la nueva versión con las nuevas funcionalidades por parte de **INTEGRANova** se realizó una segunda visita a la empresa **Aguas del Bullent S.A.** en la que se llevaron a cabo varios pasos que dieron fin a la implantación y pusieron en marcha el nuevo sistema :

- **Entrega de material** : se entregó todo el material necesario que hasta ahora no había sido entregado. Este material estaba compuesto por :
  - **Dispositivos móviles** : tres dispositivos móviles **Datalogic Blackjet™**.
  - **Accesorios de dispositivos móviles** : baterías adicionales, cables de conexión USB, adaptadores de carga de batería, peanas de carga y conexión
  - **Impresora térmica** : impresora **Zebra S4M ZPL 200 dpi** de transmisión térmica para imprimir los códigos de barras de los contadores.
  - **Fungibles de impresora** : etiquetas de material **Valéron®** y cinta de impresión.
- **Entrega de documentación** : se entregaron los manuales elaborados que explican el funcionamiento del sistema desarrollado, desde la aplicación de los dispositivos, **waterLU**, hasta las nuevas funcionalidades de **AguasCli**, pasando por la instalación de las aplicaciones y la impresora.
- **Instalación de la impresora** : se instaló en el entorno de trabajo de **Aguas del Bullent S.A.** la impresora de impresión térmica **Zebra S4M ZPL 200 dpi** de manera que estuviera correctamente integrada con el nuevo sistema.
- **Pruebas y formación** : al tiempo que se llevaban a cabo las últimas pruebas sobre el sistema se presentaba a los usuarios finales las nuevas funcionalidades con ejemplos reales de uso de manera que se familiarizaran pronto con el sistema e intentando resolver las dudas que surgieron.

Con el buen término de este último paso el nuevo proceso de toma de lecturas queda integrado en el sistema de **Aguas del Bullent S.A.**

### 7.2.1.- Desarrollos adicionales a la implantación

Con el fin de facilitar la transición del sistema antiguo al nuevo se desarrollaron una serie de documentos de uso e instalación y un instalador de la aplicación **waterLU** en dispositivos móviles.

- **Documentación y manuales** : se desarrollaron dos documentos, uno para la aplicación de los dispositivos **waterLU** y otro para la aplicación de escritorio **AguasCli** :
  - **Manual de waterLU** : explica el uso de la nueva aplicación desarrollada para los dispositivos, detallando la navegación por la interfaz, el proceso de sincronización, el uso del teclado de los dispositivos móviles y la instalación de la aplicación tanto en dispositivos **Datalogic Blackjet** como en dispositivos generales.
  - **Manual de AguasCli** : explica el uso de las nuevas funcionalidades

añadidas a la parte cliente de la aplicación, detallando el proceso de sincronización, la configuración de los dispositivos móviles, la impresión de códigos de barras y la instalación de la impresora **Zebra S4M ZPL 200 dpi**.

Estos manuales pueden verse en los anexos del documento.

- **Instalador** : se desarrolló un instalador de la aplicación para dispositivos con **Windows CE**, de forma que la aplicación pueda instalarse o reinstalarse fácilmente en un dispositivo.





## CAPÍTULO 8- Conclusiones

En este proyecto se ha dado solución informática a la problemática que planteaba el proceso de toma de lecturas de contadores en la empresa **Aguas del Bullent S.A.** Desde el principio del proyecto se ha pretendido conseguir tres objetivos en la solución.

El primero de ellos era la automatización de la tarea en todas las áreas posibles, con el fin de reducir el coste temporal de llevar a cabo las lecturas y eliminar los errores provocados por la intervención humana. Las tareas más costosas y en las que la intervención de los operarios provocaba la introducción de errores han sido automatizadas y controladas por el sistema informático. Estas tareas eran : **1)** la identificación de contadores, que ahora se realiza mediante la lectura de códigos de barras sin lugar a error, **2)** la introducción de lecturas de contadores, que ahora es controlada con valores de previsión y rangos de valor correcto basados en análisis estadísticos del historial de consumo, y **3)** la sincronización de datos entre dispositivos y servidor central, que ahora se realiza automáticamente, registrando en el servidor los datos tal y como han sido recogidos por los dispositivos.

El segundo objetivo aspira a conseguir una buena usabilidad en la aplicación del dispositivo. El perfil del usuario final de la aplicación desarrollada imponía un tratamiento especial a la interfaz, rechazando los elementos de interfaz clásicos de los dispositivos móviles, reemplazándolos por nuevos elementos que : **1)** son más grandes, en el área del elemento y en el texto, **2)** más claros, con un contraste alto para una correcta visibilidad en entornos no óptimos, y **3)** eliminan la necesidad de utilizar herramientas de precisión como el *stylus*.

El tercer objetivo trata de lograr una completa integración de la nueva aplicación de dispositivo y de las nuevas funcionalidades en el sistema ya existente. La introducción del nuevo proceso de toma de lecturas y de la nueva aplicación de los dispositivos ha llevado a añadir nuevas funcionalidades en el sistema ya existente. Estas nuevas funcionalidades han sido añadidas mediante cambios o adiciones al modelo **OLIVANOVA** de la aplicación siempre que fuera posible, o desarrollando la funcionalidad a mano e incluyéndola en el código generado de forma que todas las nuevas operaciones y funcionalidades pueden controlarse y llevarse a cabo desde la misma herramienta ya existente.

Para el desarrollo de la solución se siguieron una serie de pasos secuencialmente, siguiendo el ciclo de desarrollo de software clásico.

- **Análisis del proceso :**

El primer paso fue conocer a fondo el proceso de toma de lecturas que empleaba la empresa en el momento, detectando las tareas críticas en las que la aplicación de soluciones informáticas podía aportar las mayores ventajas.

- **Propuestas y selección de soluciones :**

Una vez conocido el proceso de toma de lecturas y detectadas las tareas críticas que lo componen comenzó la búsqueda de una solución que mejorara el proceso. Se barajaron diversas opciones para cada una de las tareas, tomando en consideración sistemas de telelectura a larga y corta distancia y la introducción de dispositivos móviles como asistentes en la toma de

lecturas; sistemas de identificación basados en OCR y códigos de barras; sistemas de toma de lectura basados en OCR y en previsión de consumo y finalmente sistemas de sincronización de datos basados en servicios web y en la metodología existente.

La solución finalmente adoptada consistía en la introducción de dispositivos móviles, identificación de contadores por código de barras, el desarrollo de una aplicación asistente en la toma de lecturas y el desarrollo de nuevas funcionalidades requeridas en la aplicación existente.

- **Diseño y desarrollo de la aplicación asistente a la toma de lecturas :**

La decisión de emplear dispositivos móviles nos permite portar el sistema informático al campo de trabajo. La aplicación diseñada y desarrollada imitaba el proceso empleado hasta el momento, esto es, identificación de contador, toma de lectura y paso al siguiente contador, al tiempo que introducía la automatización de la mayoría de tareas reduciendo coste temporal y errores.

Durante todo el proceso de diseño y desarrollo se tuvo en cuenta el perfil del usuario final, afinando la interfaz y usabilidad de la aplicación desechando los controles habituales en los dispositivos móviles y reemplazándolos por nuevos diseños de menor complejidad y mayor visibilidad y acceso.

Aprovechando el modelo de datos y la organización de los contadores ya existente en la empresa se portó a los dispositivos un subconjunto del modelo que mantenía la organización al tiempo que ofrecía la información básica necesaria para la toma de lecturas. La información portada incluía datos de contadores con previsiones de consumo, datos de rutas y su organización y datos de operarios con derecho a emplear la aplicación.

- **Diseño y desarrollo de las nuevas funcionalidades de la aplicación :**

La introducción de los dispositivos móviles crea nuevos requisitos a cumplir en la aplicación existente para la completa integración del nuevo sistema. Se diseñaron e implementaron nuevas funcionalidades para la sincronización de datos entre los dispositivos y el servidor central, la impresión de códigos de barras con la identificación de los contadores sobre una impresora térmica, la configuración de los dispositivos centralizada y soporte al registro de lecturas e incidencias.

El desarrollo de estas nuevas funcionalidades ha estado marcado desde un principio tanto por el empleo de la metodología ya existente, permitiendo un acceso transparente a datos y servicios, como por la integración con la aplicación ya desarrollada y en funcionamiento, uniendo todas las funcionalidades del sistema bajo la misma aplicación, como por un empeño en el control de la carga, empleando patrones de diseño que evitan sobrecargar el servidor central en la consulta de datos o en la ejecución de servicios.

- **Desarrollo de documentación y manual :**

Al tratarse de un proyecto para el que no va a ofrecerse mantenimiento a largo plazo es fundamental proporcionar una buena documentación que permitan en un futuro realizar modificaciones al desarrollo realizado con el fin de adaptarlo a nuevas versiones de la metodología empleada, adaptación a otras plataformas o inclusión de nuevas funcionalidades etc..

Por otra parte también se han escrito dos manuales de uso e instalación de la aplicación del dispositivo **waterLU** y de las nuevas funcionalidades de la

aplicación **AguasCli**. Estos manuales están orientados al perfil de usuario final que empleará cada parte del nuevo sistema.

- **Implantación del nuevo sistema :**

Fase final del proyecto, y dividida en dos etapas. En una primera etapa, en un estadio de desarrollo inacabado pero muy avanzado, se realizó una entrevista con los usuarios finales que evaluaron y validaron el sistema. El propósito principal de la reunión fue la recogida de observaciones sobre el diseño y desarrollo realizado captando todas las sugerencias de cambio y adiciones que permitirían al sistema una mayor integración con el sistema existente, una transición menos costosa y una usabilidad apropiada.

En una segunda etapa se actualizó la aplicación existente y se visitaron de nuevo las instalaciones de **Aguas del Bullent S.A.** para proceder a la implantación final del sistema. Durante esta fase se trabajó conjuntamente con **INTEGRANOVA**, empresa mantenedora de la estructura informática de **Aguas del Bullent S.A.**, para introducir las nuevas funcionalidades en la aplicación ya existente. Una vez actualizadas las partes cliente y servidora se puso en funcionamiento el nuevo sistema con la entrega del material necesario, esto es, dispositivos y accesorios, impresora térmica, aplicaciones y manuales, y una sesión de prueba y formación en la que se demostró el funcionamiento del nuevo proceso de toma de lecturas al tiempo que se instruía a los usuarios finales.

En la ejecución de este proyecto se han conseguido aunar tecnologías de dispositivos móviles, de entornos cliente-servidor, de lectura de códigos de barras y de impresión térmica para desarrollar completamente una solución informática a un proceso de toma de lecturas de contadores integrándola plenamente en el sistema ya existente al tiempo que se colaboraba con dos empresas (**Aguas del Bullent S.A.** e **INTEGRANOVA**.)



## CAPÍTULO 9- Bibliografía y referencias

- [1] <http://www.care-t.com/products/modeler.asp>. Herramienta de modelado OLIVANova Modeller. Último acceso 29 de marzo de 2007.
- [2] <http://software.OLIVANova.com/index.aspx>. Herramienta de modelado OLIVANova Modeller. Último acceso 29 de marzo de 2007.
- [3] <http://www.badgermeter.com/orion.htm>. Sistemas de telelectura de contadores de agua de larga distancia. Último acceso 29 de marzo de 2007.
- [4] <http://www.globalw.com/products/CZ2000.html>. Sistemas de telelectura de contadores de agua de corta distancia. Último acceso 29 de marzo de 2007.
- [5] [http://www.datalogic.com/\\_vti\\_g2\\_p\\_det.aspx?idp=ID20041211110030&rpstry=17\\_](http://www.datalogic.com/_vti_g2_p_det.aspx?idp=ID20041211110030&rpstry=17_). Dispositivo móvil industrial Datalogic Blackjet. Último acceso 29 de marzo de 2007.
- [6] <http://www.veo.com/>. Cámaras adaptables a dispositivos móviles. Último acceso 29 de marzo de 2007.
- [7] <http://www.socketmobile.com/products/bar-code-scanning-data-collection/series3/>. Lectores de códigos de barras adaptables a dispositivos móviles. Último acceso 29 de marzo de 2007.
- [8] [http://en.wikipedia.org/wiki/Ingress\\_protection](http://en.wikipedia.org/wiki/Ingress_protection). Índice de protección en aplicaciones eléctricas contra objetos sólidos, polvo, contacto accidental y agua. Último acceso 29 de marzo de 2007.
- [9] <http://www.iec.ch/>. Comisión internacional de electrónica. Último acceso 29 de marzo de 2007.
- [10] <http://www.microsoft.com/sql/editions/sqlmobile/sqlmobileresources.msp>. Microsoft Sql Server 2005 Mobile. Último acceso 29 de marzo de 2007.
- [11] <http://www.microsoft.com/spanish/msdn/vs2005/default.msp>. Microsoft Visual Studio 2005. Último acceso 29 de marzo de 2007.
- [12] [http://en.wikipedia.org/wiki/Barcodes#Types\\_of\\_barcode](http://en.wikipedia.org/wiki/Barcodes#Types_of_barcode). Clasificación de tipos de códigos de barras. Último acceso 29 de marzo de 2007.
- [13] <http://www.barcodeisland.com/upca.phtml>. Estándar de código de barras UPC-A. Último acceso 29 de marzo de 2007.
- [14] <http://www.barcodeisland.com/ean13.phtml>. Estándar de código de barras EAN-13. Último acceso 29 de marzo de 2007.
- [15] <http://grandzebu.net/index.php?page=informatique/codbar/ean13.htm>. Fuente y algoritmo para la creación de códigos de barras EAN-13.
- [16] [http://www.zebra.com/id/zebra/na/en/index/products/printers/industrial\\_commercial/s4m.html](http://www.zebra.com/id/zebra/na/en/index/products/printers/industrial_commercial/s4m.html). Impresora de etiquetas por transferencia térmica Zebra S4M ZPL 200 dpi.
- [17] <http://www.materialconcepts.com/pages/valeron-strength-film.asp>. Material de alta resistencia Valéron®.
- [18] <http://www.INTEGRANova.com/>. Empresa INTEGRANova.



## CAPÍTULO 10- Anexos

### 10.1.- Manual de uso de **waterLU**

### Manual de uso e instalación de **waterLU**



- Página intencionadamente en blanco -



# Índice

I - Uso de la aplicación.....	6
I.1 - Identificación de entrada al sistema.....	7
I.2 - Menú principal.....	8
I.2.a - Con configuración disponible.....	8
I.2.b - Sin configuración disponible.....	8
I.3 - Selección de ruta.....	9
I.4 - Seguimiento de ruta.....	10
I.5 - Toma de lecturas.....	11
I.6 - Incidencias en lecturas.....	12
I.7 - Configuración general.....	13
I.7.a - Opciones de configuración:.....	13
I.8 - Sincronización.....	15
I.8.a - Proceso de sincronización.....	15
I.9 - Configuración de sincronización.....	17
I.9.a - Operaciones de sincronización.....	18
I.9.b - Tareas típicas de sincronización.....	18
I.10 - Pantalla de bloqueo.....	19
I.11 - Abandonar la aplicación.....	20
I.12 - Cambio de la contraseña de salida de la aplicación.....	21
I.13 - Avisos / Confirmaciones.....	22
II - Adaptación de la aplicación al dispositivo Datalogic Blackjet™.....	26
II.1 - Uso del teclado.....	26
II.1.a - Teclado del dispositivo Datalogic Blackjet™.....	26
II.1.b - Teclas especiales.....	26
II.2 - Notas de navegación.....	27
II.2.a - En todas las pantallas.....	27
II.2.b - Campos de entrada de datos.....	28
II.2.c - Cursores para navegación en rutas.....	28
III - Instalación de la aplicación.....	30
III.1 - Requisitos.....	30
III.2 - Instalación en dispositivo Datalogic BlackJet™.....	30
III.2.a - Instalación general.....	32
III.3 - Instalación de requisitos.....	34
III.3.a - Identificación del instalador para un dispositivo y sistema.....	36
IV - Tras la instalación.....	37
V - Importante.....	38

- Página intencionadamente en blanco -

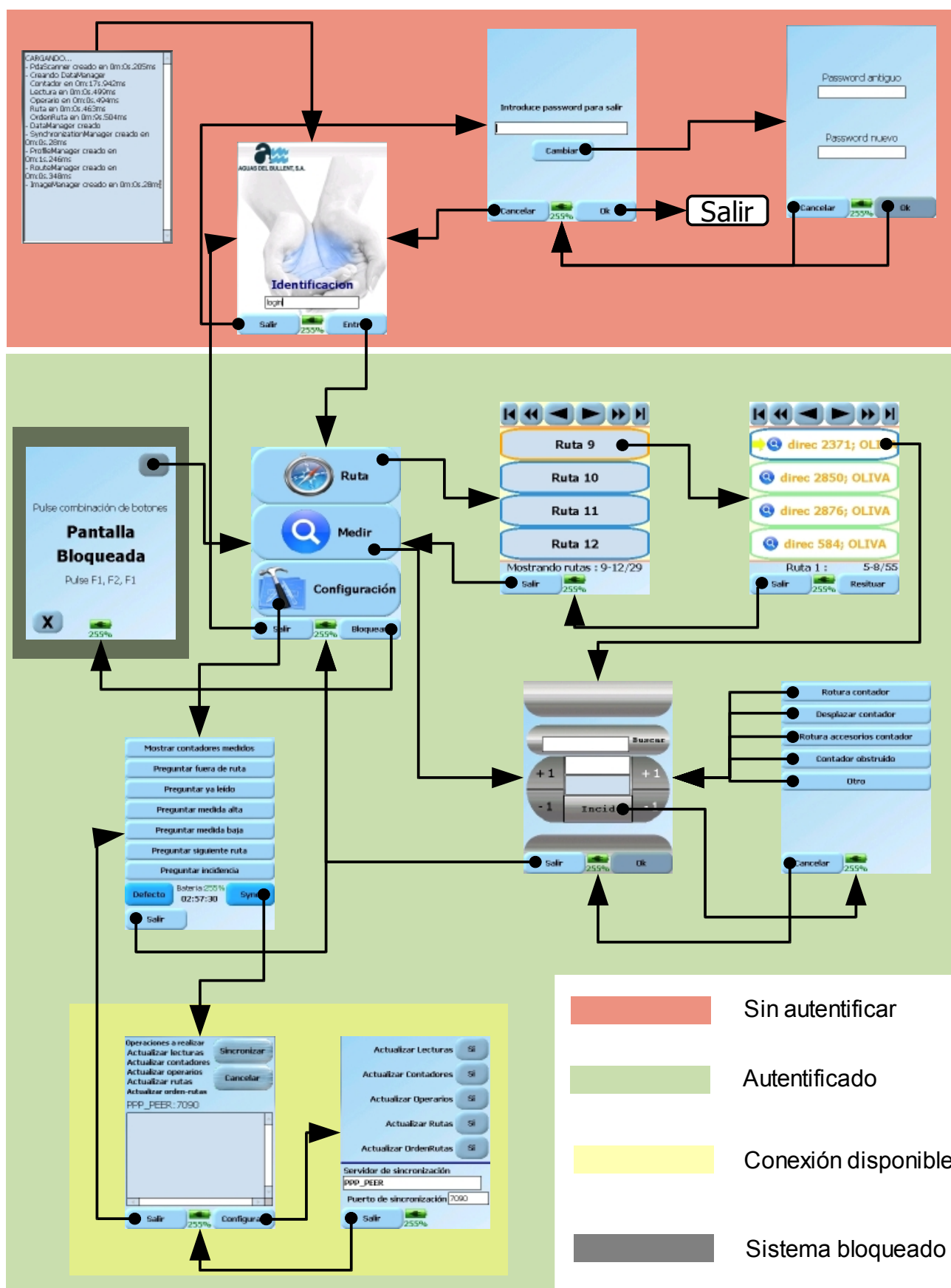
Este manual asume:

- que el usuario está familiarizado con el uso de aplicaciones que interaccionan mediante botones y campos de entrada de datos.
- que el usuario sabe como conectar el dispositivo a un equipo mediante *ActiveSync*, o bien activar las capacidades WiFi o similar que permitan acceder a la red.
- que el usuario está familiarizado con el empleo de pantallas táctiles.
- que dispone de la última versión de **waterLU**.
- que el usuario tiene acceso tanto al manual de nuevas funcionalidades de **AGUASCLI** como a la aplicación **AGUASCLI**.

Este manual emplea estos términos/generalidades:

- **dispositivo** o **dispositivo móvil**: cualquier dispositivo, ya sea una PDA, un SmartDevice, PocketPC, etc. sobre el que puede ejecutarse **Windows CE 5.0**, y por ende, esta aplicación.
- **lectura** o **medida** : el hecho de tomar la lectura de consumo de un contador o el propio valor de consumo de un contador.
- **AGUASCLI** : programa que es la parte cliente del sistema generado con **OLIVANOVA**. Es la aplicación que se ejecuta sobre un computador de sobremesa.
- **servidor** o **servidor central** : programa que es la parte servidora del sistema generado con **OLIVANOVA**. Es la aplicación que se ejecuta en el servidor.
- **conectividad** o **disponer de conexión** : se produce cuando, o bien el dispositivo está conectado a un equipo mediante *ActiveSync*, o bien está conectado a un red mediante WiFi, Bluetooth, etc.

# 1 Uso de la aplicación



## 1.1 Identificación de entrada al sistema.

Para poder llevar a cabo tareas con la aplicación, es necesario identificarse ante el sistema. Esta identificación se realiza en la pantalla de **Identificación**. Tan sólo es necesario introducir el código de identificación, y si es

El diagrama ilustra la interfaz de identificación de la aplicación AGUAS DEL BULLENT, S.A. La interfaz tiene un fondo azul claro con una imagen de manos sosteniendo un objeto. En la parte superior izquierda, se encuentra el logo de la empresa. El título principal es "Identificación". Debajo de este, hay un campo de texto con el placeholder "login". A la izquierda del campo de texto hay un botón "Salir" y a la derecha un botón "Entrar". Entre ambos botones hay un indicador de progreso que muestra "94%".

El diagrama muestra dos flujos de uso:

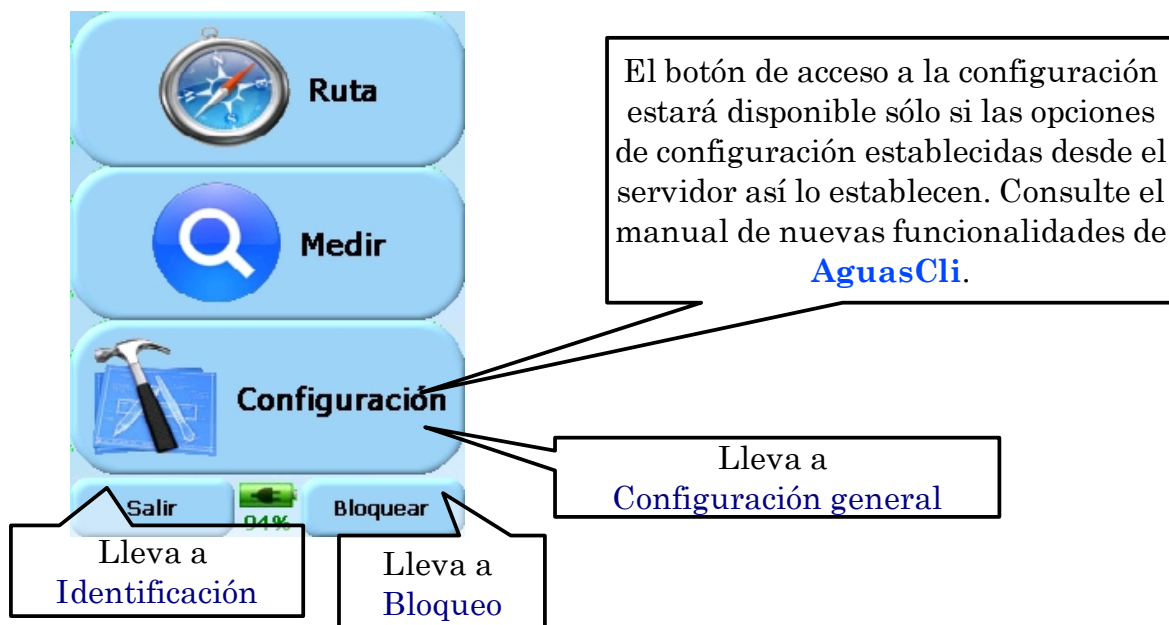
- Flujo de éxito:** Se muestra la interfaz con el campo de texto y los botones habilitados. Un mensaje indica: "correcto se tendrá acceso a la aplicación." Los botones están etiquetados: "Salir" lleva a "Abandonar la aplicación" y "Entrar" lleva a "Menú principal".
- Flujo de error:** Se muestra la interfaz con el campo de texto y los botones deshabilitados (gris). Un mensaje indica: "En el caso de introducir un código de autenticación que no es reconocido por el sistema, la pantalla se enrojece mostrando el error, y durante unos segundos se deshabilitan los botones y el campo de entrada de código. Una vez la pantalla vuelve a su color azul, se habilitan de nuevo el campo de entrada de código y los botones." Otro mensaje indica: "Al ocurrir un error en la identificación, la pantalla de identificación se deshabilita unos segundos."

Los mensajes de error y éxito están presentados en cuadros de texto con líneas que indican su posición en la interfaz.

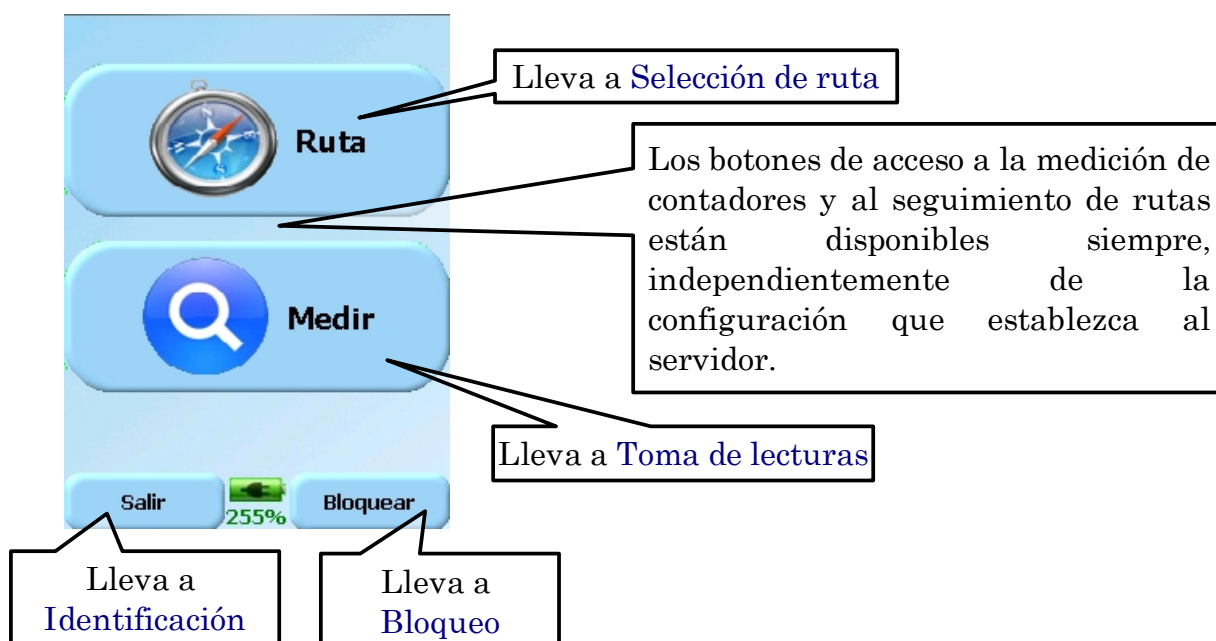
## 1.2 Menú principal

El menú principal de la aplicación tiene dos vistas posibles, dependiendo de la configuración que se haya establecido. Las vistas únicamente difieren en la posibilidad de acceso a las opciones de configuración de la aplicación. El seguimiento de rutas y la medición de contadores siempre está disponible, independientemente de la configuración.

### 1.2.1 Con configuración disponible

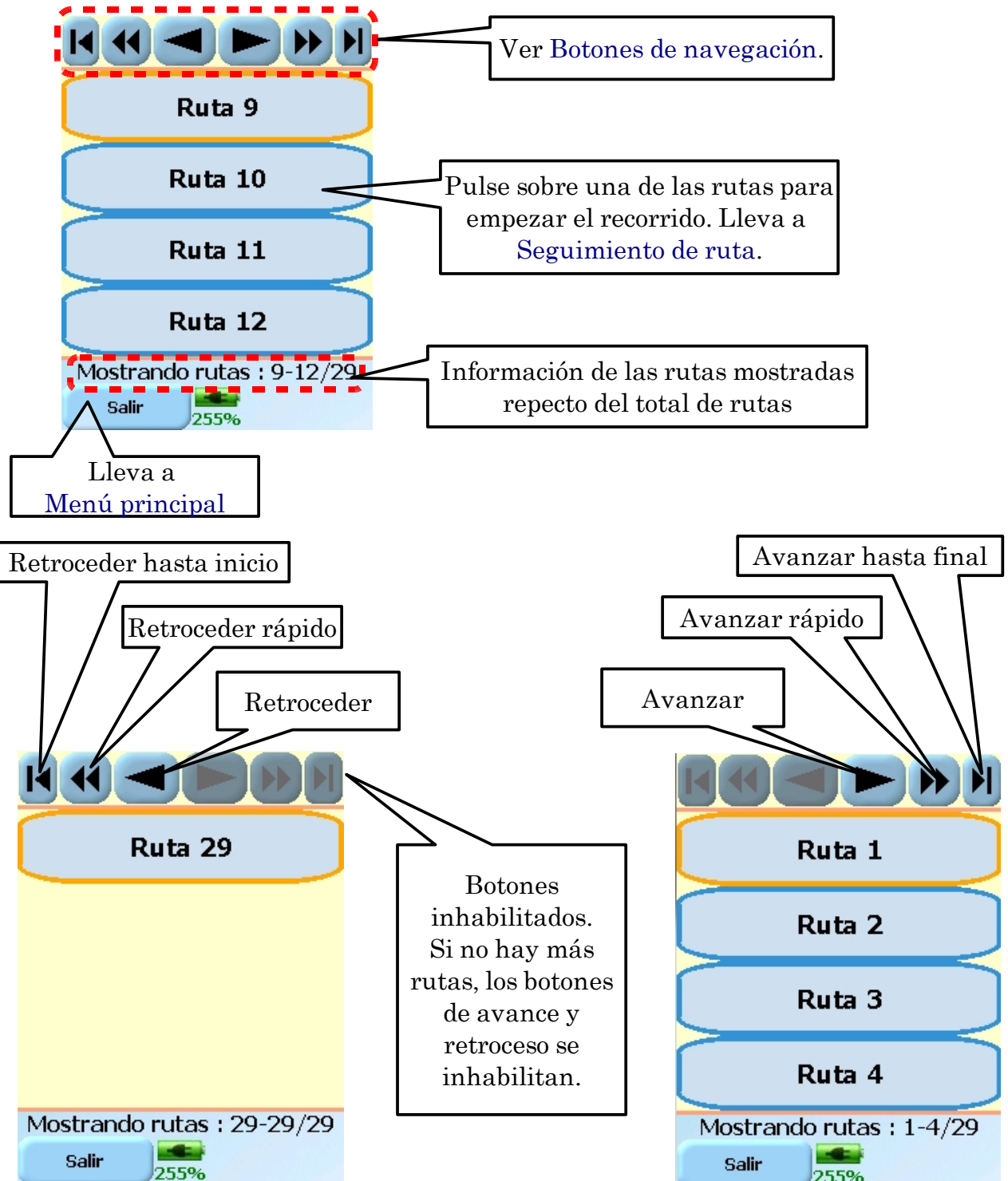


### 1.2.2 Sin configuración disponible



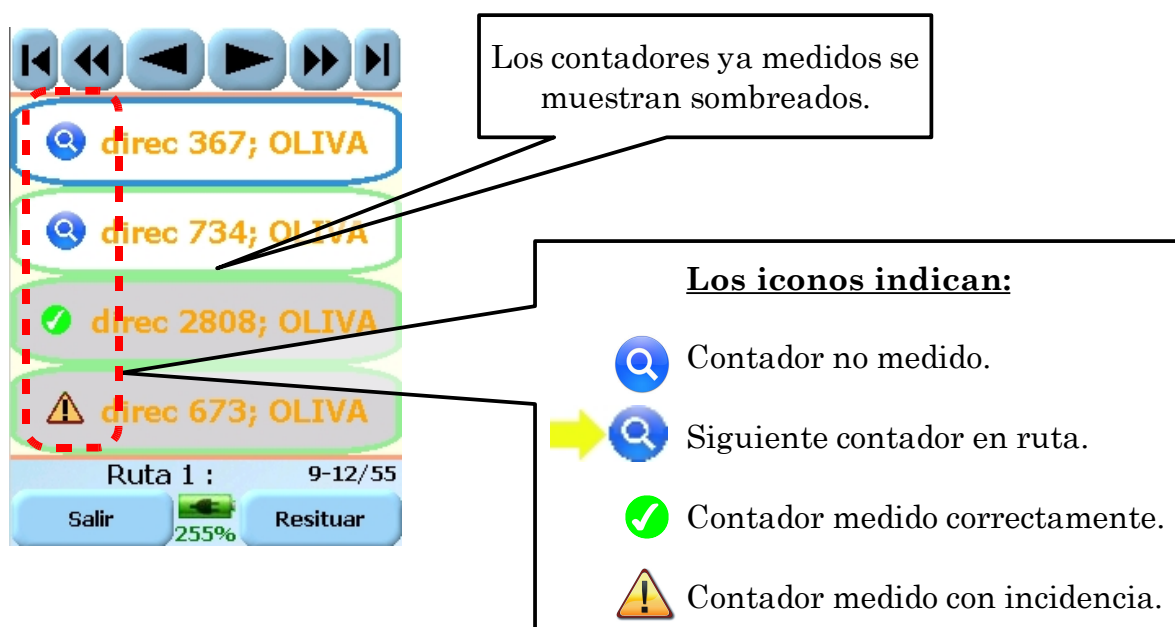
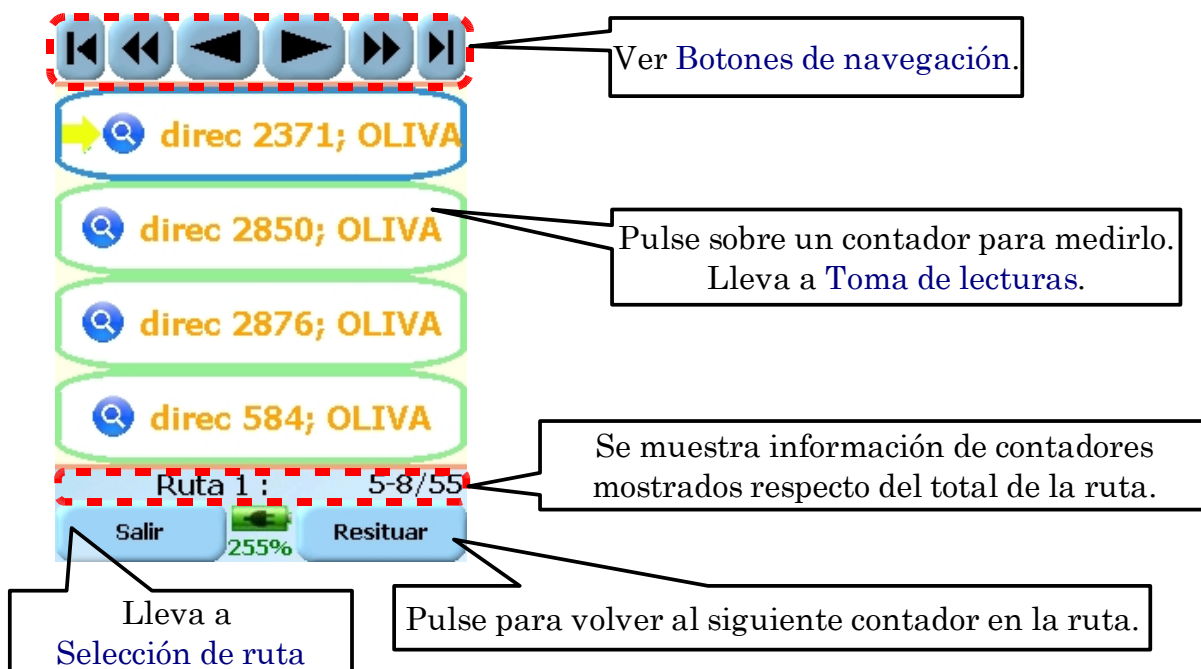
### 1.3 Selección de ruta

La pantalla de selección permite seleccionar la ruta a seguir para realizar su seguimiento. El seguimiento de una ruta se utiliza como guía indicativa de cuál es el siguiente contador que se va a medir. Según la configuración que tenga establecida la aplicación puede permitirse medir contadores que se encuentren fuera de ruta.



## 1.4 Seguimiento de ruta

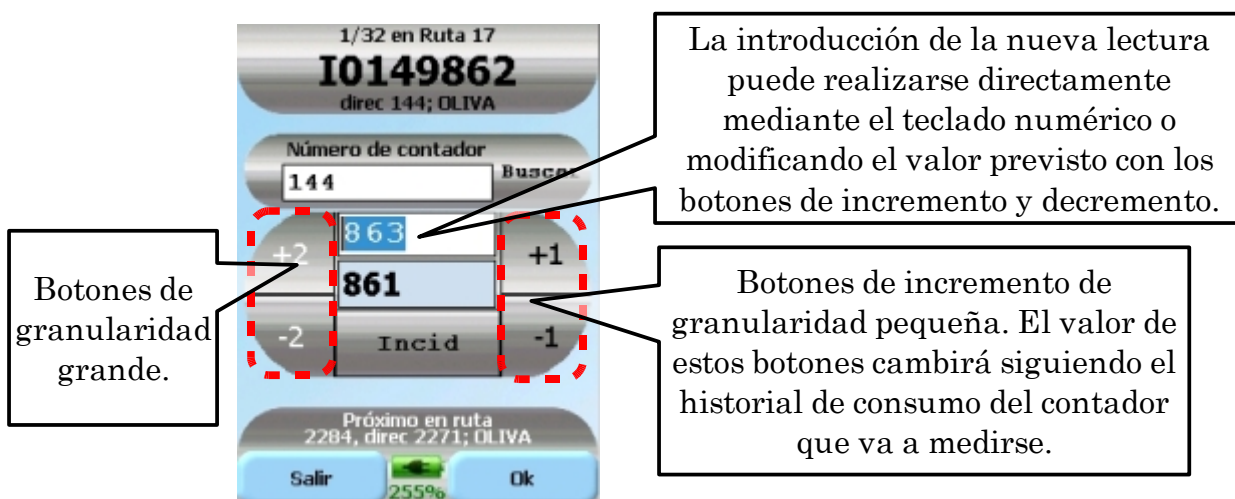
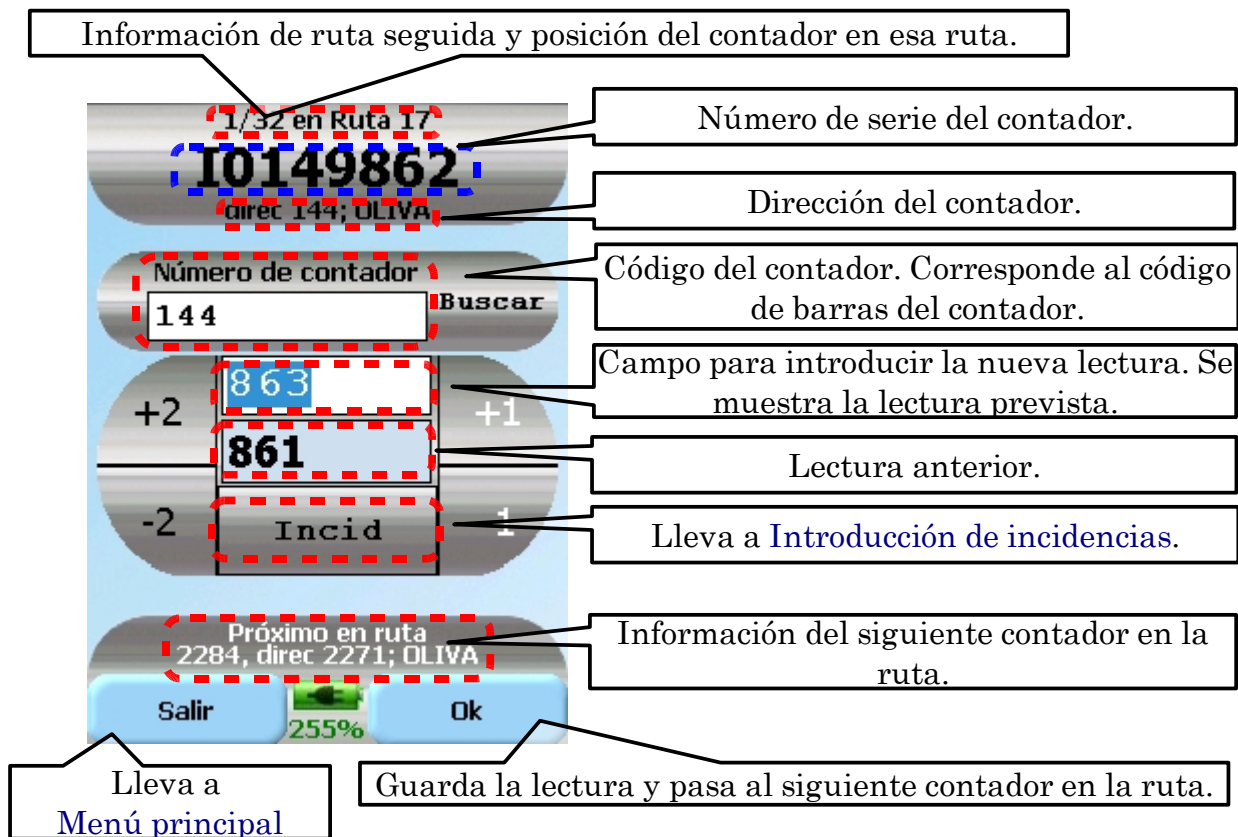
La pantalla de seguimiento de ruta muestra información relativa a la ruta que se está siguiendo. Se utilizan iconos para representar el estado de cada contador en la ruta. Dependiendo de la configuración establecida en la aplicación pueden o no mostrarse los contadores ya medidos. Ofrece una interfaz de navegación similar a la pantalla de selección de ruta.






## 1.5 Toma de lecturas

La pantalla de toma de lecturas permite la introducción de medidas de consumo de contadores, siguiendo o no una ruta. Se muestra información relativa a la ruta seguida y posición en la ruta, el número de serie del contador, la dirección del contador, el número de contador (correspondiente al número del código de barras), la lectura prevista, la lectura anterior e información del siguiente contador en la ruta.



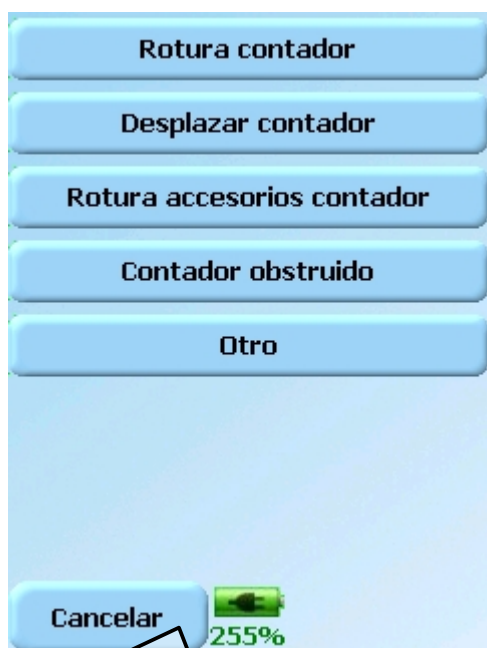
Si se dispone de un teclado con cursores  pueden utilizarse para seleccionar la granularidad con derecha e izquierda, e incrementar y decrementar con arriba y abajo.

## 1.6 Incidencias en lecturas

En la lectura de un contador puede registrarse una incidencia, que será guardada por la aplicación y transmitida en el momento de la sincronización. La posibilidad de registrar incidencias está basada en la configuración establecida en la aplicación.




Durante la toma de una lectura, pulse el botón **Incid** para registrar una incidencia.



Lleva a [Toma de lecturas](#).  
Cancela el registro de incidencia.

Pulse el tipo de incidencia que se ha producido.  
Se registrará automáticamente y se pasará a medir el siguiente contador de la ruta.

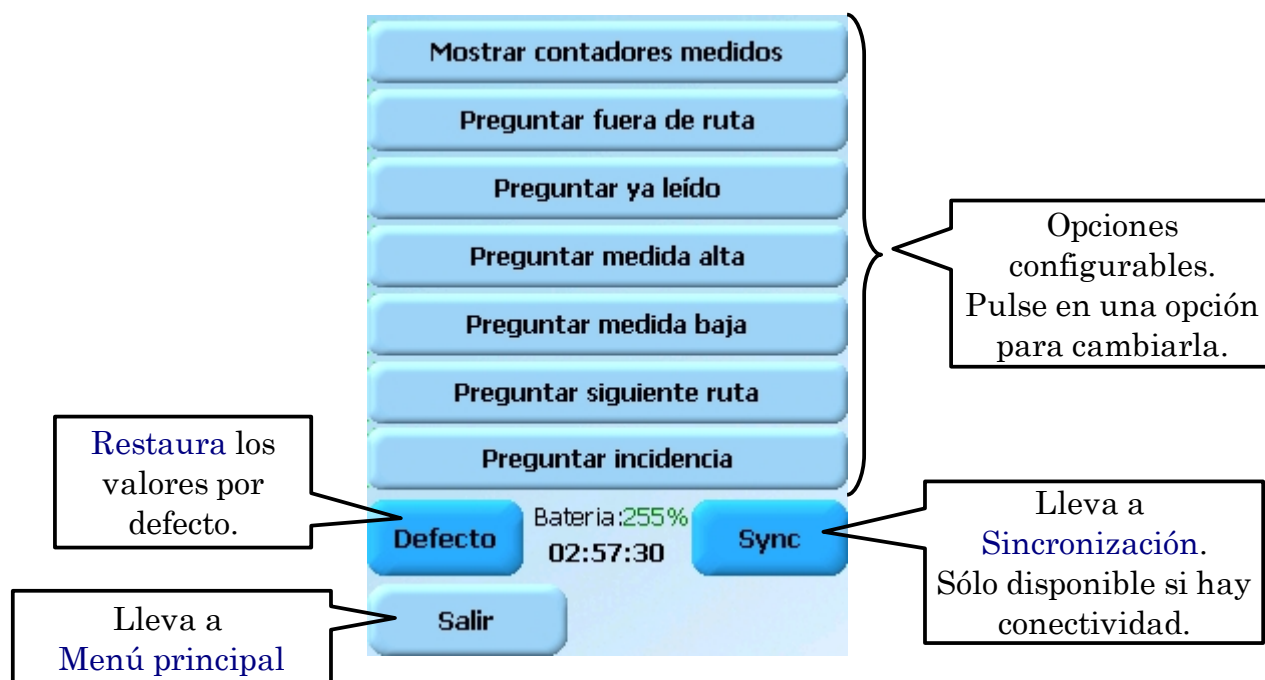
Sólo puede registrarse una incidencia por contador.

Las contadores para los que se ha registrado una incidencia se muestran en la pantalla de [Seguimiento de ruta](#) con este icono : 

Las incidencias quedan registradas en la aplicación hasta la próxima sincronización, momento en el que pasan al servidor para que sean tratadas.

## 1.7 Configuración general

En la pantalla de configuración puede establecerse el comportamiento que tendrá la aplicación mientras esté en funcionamiento. El valor que se establezca para cada una de estas opciones determinará que la aplicación permita, pida confirmación o directamente prohíba algunas acciones durante su ejecución. Esta pantalla estará disponible solo si la configuración que el servidor establece sobre la aplicación lo permite. Consulte el manual de nuevas funcionalidades de [AGUASCLI](#).



### 1.7.1 Opciones de configuración:

[D] Indica la opción por defecto que se establecerá al restaurar los valores por defecto.

- [Mostrar / Esconder] contadores ya medidos :

**Mostrar contadores medidos**

En la pantalla de navegación de ruta se muestran los contadores ya medidos.

[D] **Esconder contadores medidos**

En la pantalla de navegación de ruta **no** se muestran los contadores ya medidos.

- [Permitir / Preguntar / Prohibir] fuera de ruta :

**Permitir fuera de ruta**

Se permite medir contadores que están fuera de la ruta seguida.

[D] **Preguntar fuera de ruta**

Se preguntará antes de medir contadores fuera de la ruta seguida.

**Prohibir fuera de ruta**

No se permitirá medir contadores fuera de la ruta seguida.

- [Permitir / Preguntar / Prohibir] medir contador ya medido :
 

	<b>Permitir remedir</b>	Se permite medir contadores ya medidos.
<b>[D]</b>	<b>Preguntar ya leído</b>	Se preguntará antes de medir un contador ya medido.
	<b>Prohibir remedir</b>	No se permitirá medir un contador ya medido.
- [Permitir / Preguntar / Prohibir] medidas demasiado altas :
 

	<b>Permitir medida alta</b>	Se permite tomar medidas demasiado altas.
<b>[D]</b>	<b>Preguntar medida alta</b>	Se preguntará antes de tomar medidas demasiado altas.
	<b>Prohibir medida alta</b>	No se permitirá tomar medidas demasiado altas.
- [Permitir / Preguntar / Prohibir] medidas demasiado bajas :
 

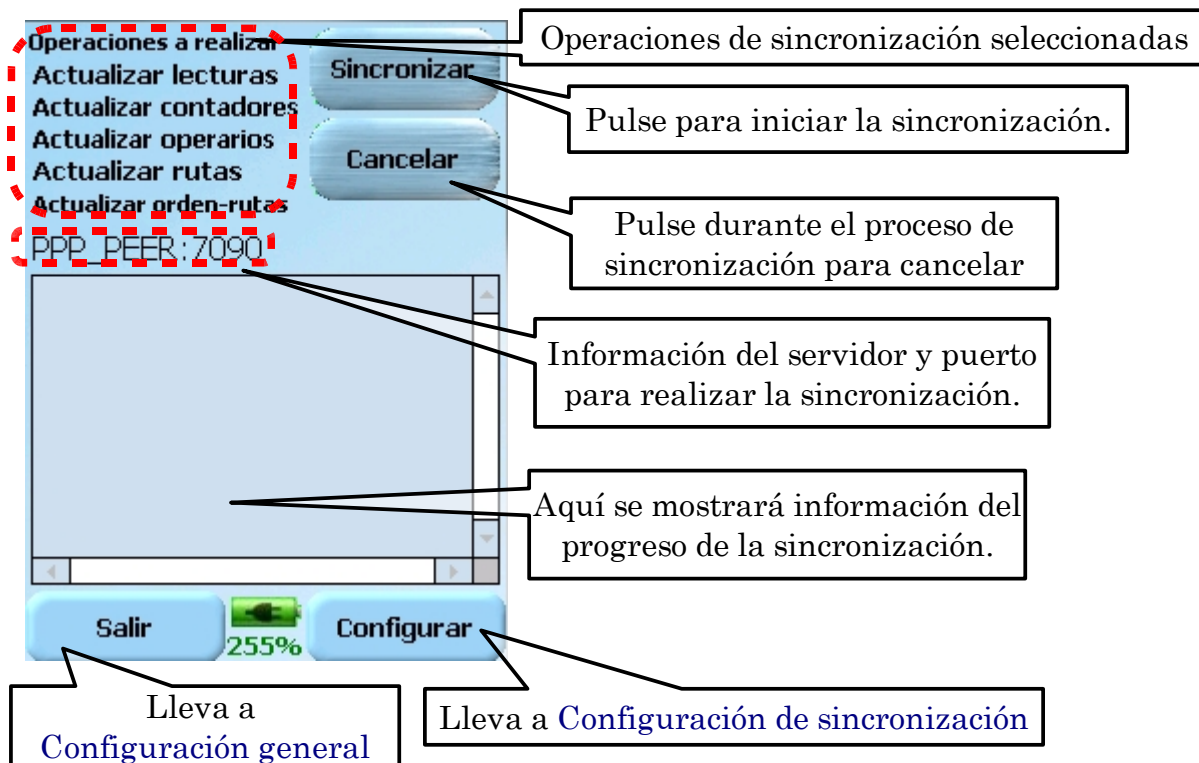
	<b>Permitir medida baja</b>	Se permite tomar medidas demasiado bajas.
<b>[D]</b>	<b>Preguntar medida baja</b>	Se preguntará antes de tomar medidas demasiado bajas.
	<b>Prohibir medida baja</b>	No se permitirá tomar medidas demasiado bajas.
- [Permitir / Preguntar / Prohibir] pasar a siguiente ruta :
 

	<b>Permitir siguiente ruta</b>	Cuando se mida el último contador de una ruta, se pasará al primero de la ruta siguiente.
<b>[D]</b>	<b>Preguntar siguiente ruta</b>	Cuando se mida el último contador de una ruta, se preguntará para pasar al primero de la siguiente ruta.
	<b>Prohibir siguiente ruta</b>	Cuando se mide el último contador de una ruta, se mostrará un aviso informando de que la ruta terminó.
- [Permitir / Preguntar / Prohibir] registrar incidencia :
 


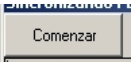
	<b>Permitir incidencia</b>	Se permite la introducción de incidencias en las lecturas de contadores.
<b>[D]</b>	<b>Preguntar incidencia</b>	Se preguntará antes de introducir una incidencia en la lectura de contadores.
	<b>Prohibir incidencia</b>	No se permitirá la introducción de incidencias en las lecturas de contadores.

## 1.8 Sincronización

La pantalla de sincronización permite descargar las lecturas tomadas en el servidor y actualizar la información de contadores, rutas, operarios y configuración en la aplicación. Se muestra información de las operaciones de sincronización a realizar, el servidor y puerto contra el que se realizarán e información del avance de todo el proceso. Esta pantalla sólo estará disponible si la aplicación dispone de conexión.



### 1.8.1 Proceso de sincronización

- 1) El primer paso debe ser configurar las operaciones a realizar durante la sincronización en [Configuración de sincronización](#).
- 2) La aplicación de escritorio debe estar iniciada y en la ventana principal.
- 3) Pulse el botón  para comenzar.
- 4) Se pedirá confirmación del proceso de sincronización en la aplicación de escritorio.
- 5) Pulse  en [AguasCli](#) para empezar.
- 6) El proceso de sincronización comienza y se muestra el progreso de las operaciones mientras se realizan.

<Conexión correcta>  
#Actualizar lecturas iniciada#  
· 11 lecturas enviadas

El proceso de sincronización ha comenzado. La conectividad debe mantenerse hasta que se realice la desconexión.

```
#Actualizar lecturas iniciada#
· 11 lecturas enviadas
#Actualizar lecturas terminada#
#Actualizar contadores iniciada#
· 3008 contadores enviados
· 0 contadores a borrar recibidos
· 3008 contadores a actualizar recibidos
· 0 contadores a añadir recibidos
```

Se muestra el progreso de las operaciones conforme se realizan. El tiempo de realización de las operaciones dependerá de la cantidad de datos a sincronizar.

- 7) Una vez se produce la desconexión, ya no es necesaria ni la aplicación de escritorio ni la conexión.

```
#Actualizar configuracion iniciada#
#Actualizar configuracion terminada#
<Desconexión correcta>
```

- 8) La aplicación comenzará a actualizar su base de datos.

```
#Actualizando tablas bd#
· Actualizando tablas bd
· Contadores actualizados
· Operarios actualizados
· Rutas actualizadas
· Orden-rutas actualizados
· Configuración actualizada
#Actualización de bd terminada#
```

- 9) Cuando todo el proceso termine, se muestra un resumen del tiempo invertido.

```
#duración: 4m:26s.617ms#
```

El tiempo que cuesta realizar un proceso de sincronización dependerá de las operaciones que se hayan seleccionado, la cantidad de datos que sea necesario sincronizar y el tipo de conexión que se utilice. **El proceso completo puede durar varios minutos.**

En cualquier momento de la sincronización el proceso puede cancelarse pulsando **Cancelar**. Si el proceso es cancelado, no se realizará ningún cambio en los datos de la aplicación.

```
#Actualizar lecturas terminada#
#Actualizar contadores iniciada#
· 3008 contadores enviados
Sincronización cancelada
```

Durante el proceso de sincronización pueden producirse errores que impidan que termine correctamente (pérdida de la conexión, fallo en el servidor, ...). Si se produce un error no recuperable, la sincronización será cancelada y no se producirán cambios en la aplicación.

```
EXCEPCIÓN: Error recibiendo mensaje.
Sincronización cancelada
Cambios desechados
#duración: 0m:2s.3ms#
```

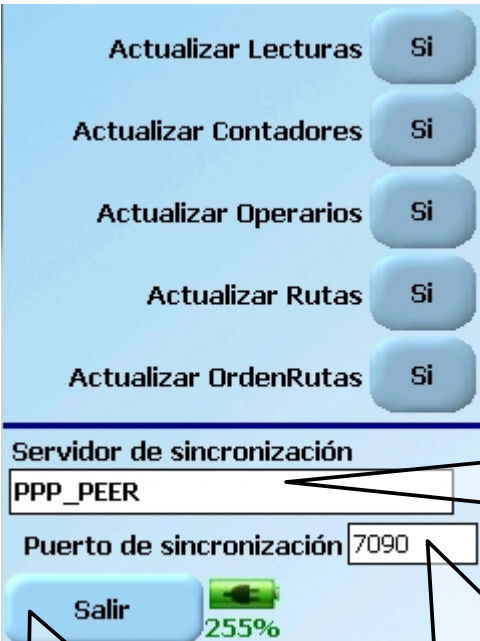
Se muestra información del error.

Al ocurrir un error grave, la sincronización se cancela y los cambios se desechan.



## 1.9 Configuración de sincronización

En la pantalla de configuración de sincronización puede configurarse el proceso de sincronización, seleccionando las operaciones a realizar y el servidor y puerto contra el que realizar la sincronización. Estas opciones sólo podrán cambiarse si la configuración de la aplicación lo permite.





The screenshot shows a configuration window with the following elements:

- Five buttons for operations: **Actualizar Lecturas**, **Actualizar Contadores**, **Actualizar Operarios**, **Actualizar Rutas**, and **Actualizar OrdenRutas**. Each button has a **Si** (Yes) label.
- Fields for **Servidor de sincronización** (containing **PPP\_PEER**) and **Puerto de sincronización** (containing **7090**).
- A **Salir** (Exit) button and a green progress indicator showing **255%**.

Annotations and callouts:

- Operation Buttons:** Pulse los botones **Si/No** para determinar si una operación se realizará durante el proceso de sincronización. Para iniciar un proceso de sincronización es necesario tener seleccionada al menos una operación.
- Servidor de sincronización:** Nombre del ordenador que hará de servidor de sincronización. Este ordenador deberá estar ejecutando la aplicación cliente, **AguasCli**, para poder realizar la sincronización.
- Puerto de sincronización:** Puerto que utiliza el servidor para realizar la sincronización.
- Salir button:** Lleva a **Sincronización**.
- General Note:** Si no se permite cambiar la configuración, los botones y campos de entrada de datos estarán inhabilitados.
- Disabled State:** Los botones de selección de operación aparecen con el valor que tienen establecido, pero están inhabilitados, impidiendo su cambio.
- Disabled Fields:** El nombre del servidor y el puerto aparecen sombreados, mostrando los valores que tienen establecidos pero impidiendo su cambio.

### 1.9.1 Operaciones de sincronización

Cada operación de sincronización puede establecerse cómo  o cómo .



La operación se realizará.



La operación no se realizará.

Las operaciones que pueden realizarse durante una sincronización son éstas:

- **Actualizar Lecturas:** Esta operación descarga sobre el servidor todas las lecturas e incidencias tomadas desde la última sincronización.
- **Actualizar Contadores:** Esta operación actualiza los datos de los contadores sobre la aplicación. Elimina los contadores dados de baja, añade los nuevos contadores dados de alta y actualiza los contadores que han cambiado.
- **Actualizar Operarios:** Esta operación actualiza los datos de los operarios que pueden identificarse y utilizar la aplicación. Elimina los operarios dados de baja, añade los nuevos operarios y actualiza los operarios que han cambiado.
- **Actualizar Rutas:** Esta operación actualiza las rutas que pueden seguirse en la aplicación. Elimina las rutas borradas, añade las nuevas rutas y actualiza las rutas cambiadas.
- **Actualizar Orden-Rutas:** Esta operación actualiza los contadores que forman cada ruta sobre la aplicación. Elimina los contadores que se quitan de su ruta, añade los nuevos contadores a la ruta correspondiente y actualiza los contadores cuyo orden en la ruta ha cambiado.
- **Actualizar Configuración:** Esta operación no es seleccionable, pero siempre se produce al final de una sincronización. De esta forma la aplicación se mantiene configurada conforme indique el servidor.

### 1.9.2 Tareas típicas de sincronización

- **Antes de la toma de lecturas:** Antes de comenzar el período de toma de lecturas es recomendable realizar una sincronización con las operaciones: *Actualizar Contadores*, *Actualizar Operarios*, *Actualizar Rutas* y *Actualizar Orden-Rutas*. De esta forma la aplicación dispondrá de los últimos datos actualizados para llevar a cabo las lecturas.
- **Una vez tomadas todas las lecturas:** Una vez ha terminado el período de toma de lecturas, es recomendable realizar una sincronización con la operación *Actualizar Lecturas*, de forma que las lecturas e incidencias se descarguen al servidor y puedan ser facturadas y tratadas.

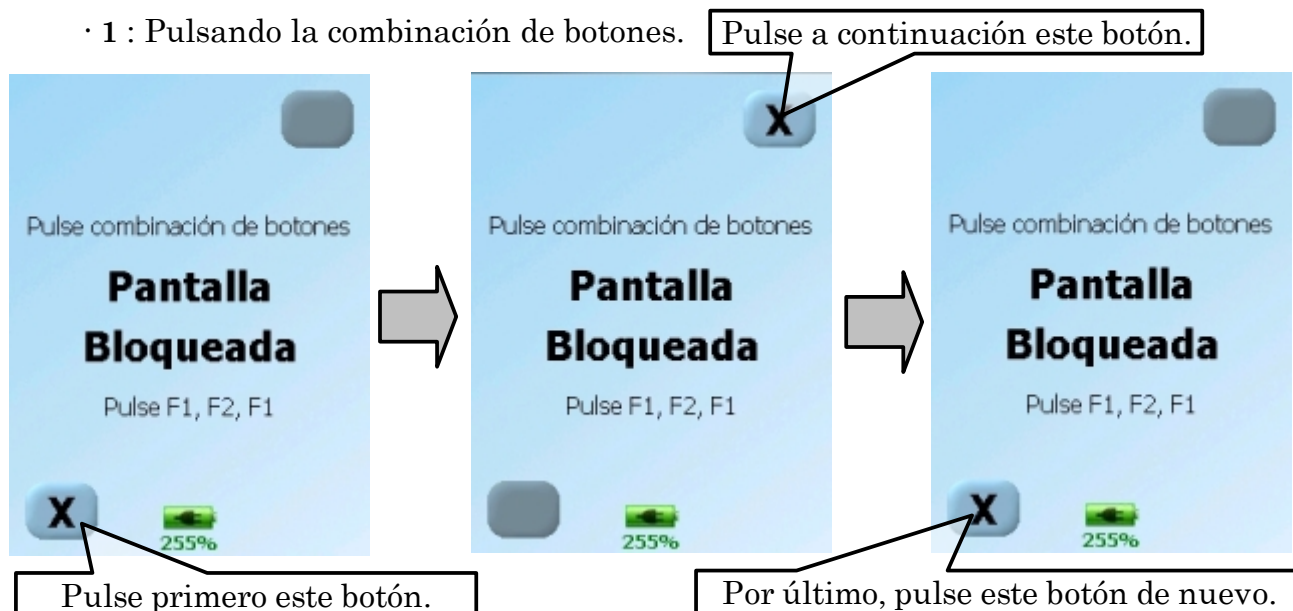


## 1.10 Pantalla de bloqueo

La aplicación dispone de una pantalla de bloqueo en la que toda interacción queda inhabilitada hasta pulsar la combinación que desactiva esta pantalla. Esta pantalla permite que en períodos en que no está empleándose el sistema se produzcan accidentalmente lecturas, cambios en la configuración, cambios de ruta, o cualquier interacción no deseada.

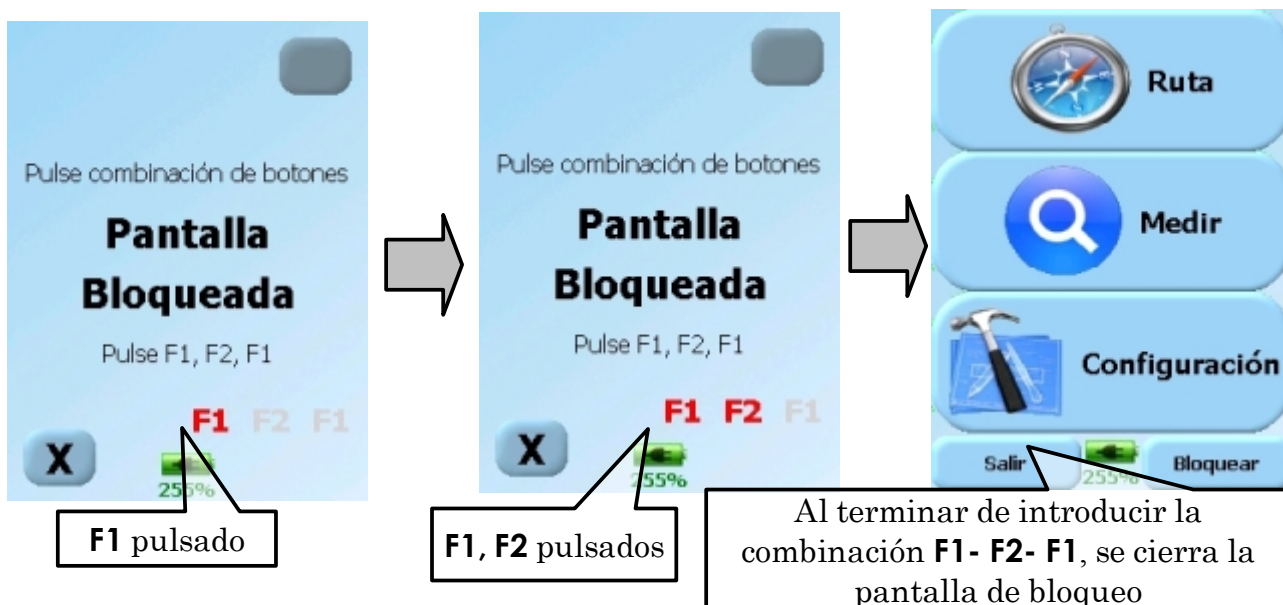
Existen dos formas de salir de la pantalla de bloqueo:

· 1 : Pulsando la combinación de botones.



· 2 : Pulsando combinación de teclas. Consulte [Teclado del dispositivo](#).

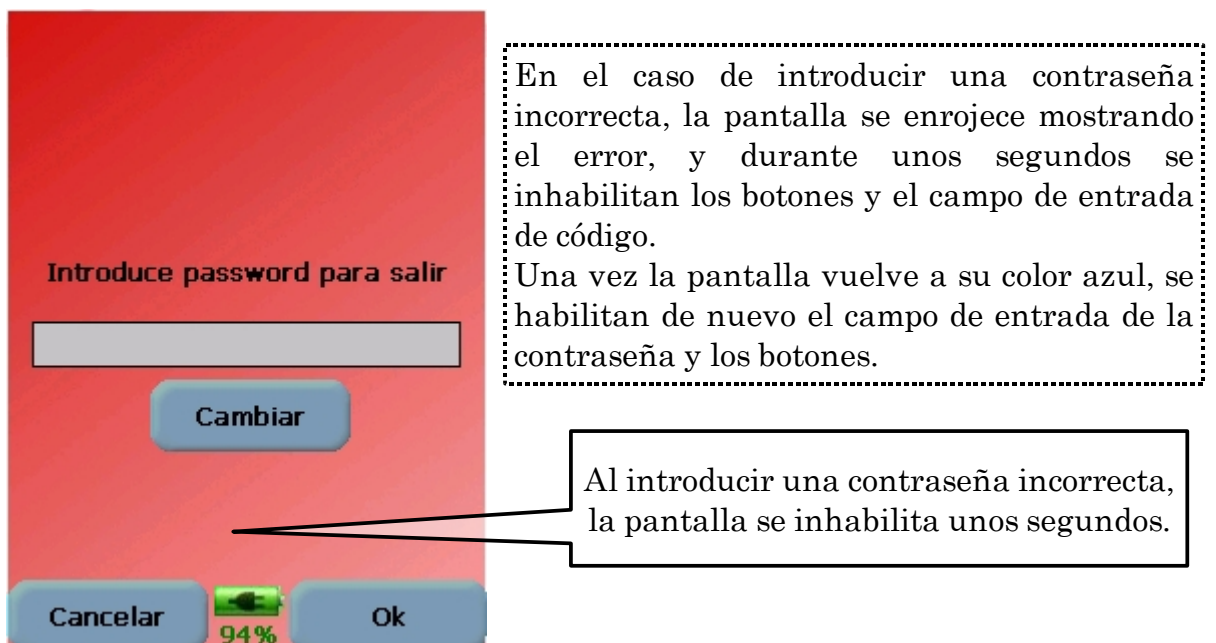
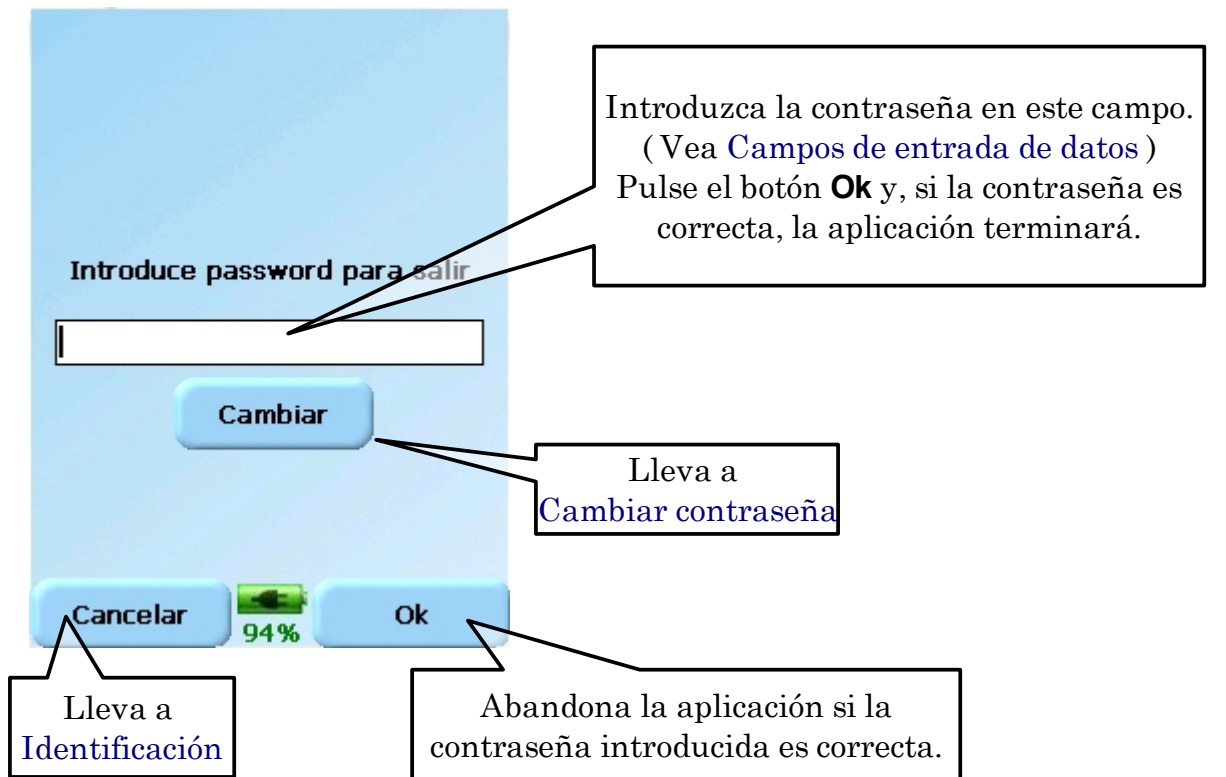
Alternativamente, puede salir de la pantalla pulsando **F1, F2, F1**



Tras utilizar uno de los métodos la pantalla de bloqueo se cerrará y podrá continuar con el funcionamiento normal de la aplicación.

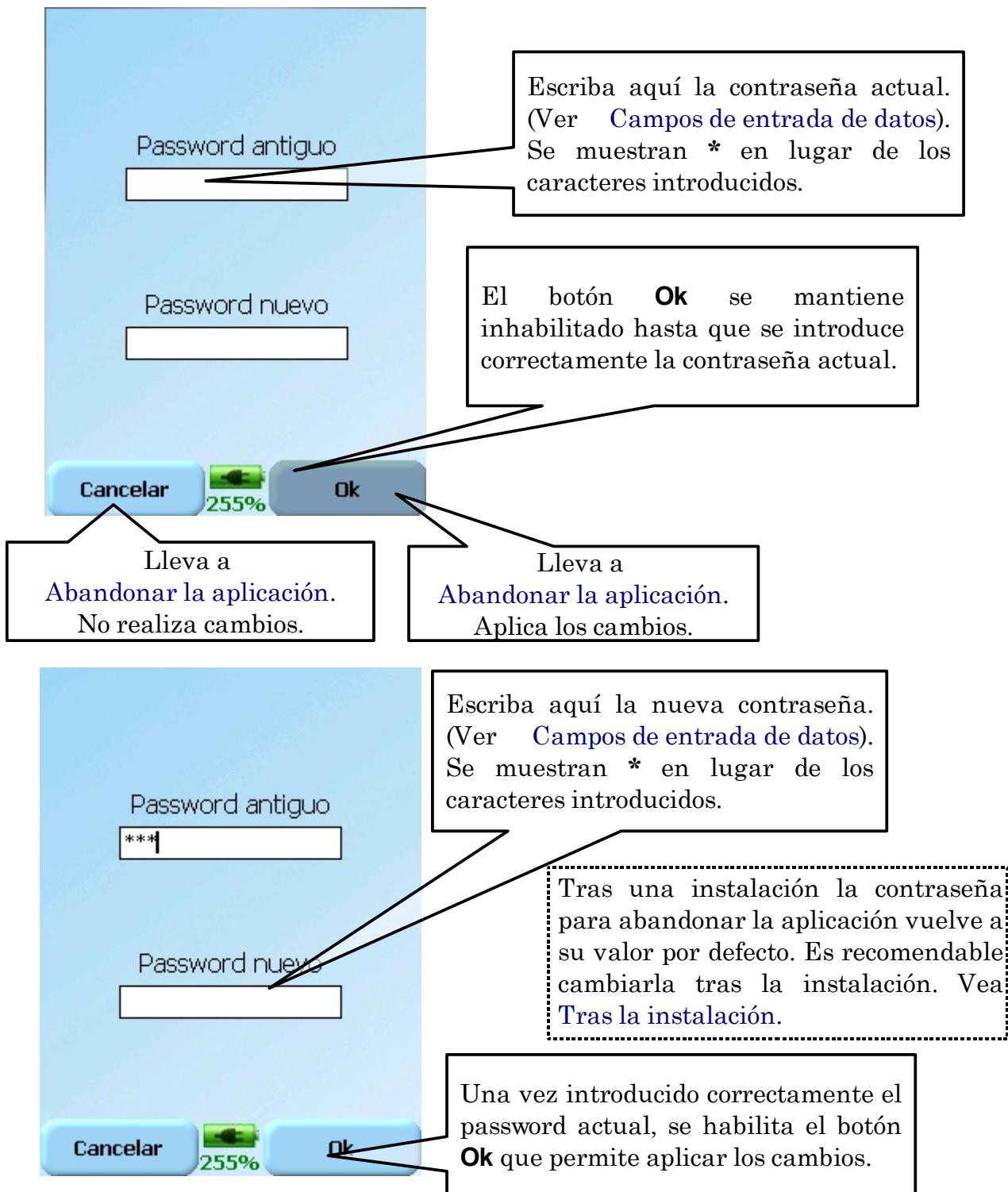
## 1.11 Abandonar la aplicación

La aplicación está diseñada para que no pueda abandonarse sin tener permiso para ello. Es por ésto que con tal de salir de la aplicación es necesario proporcionar una contraseña.



## 1.12 Cambio de la contraseña de salida de la aplicación

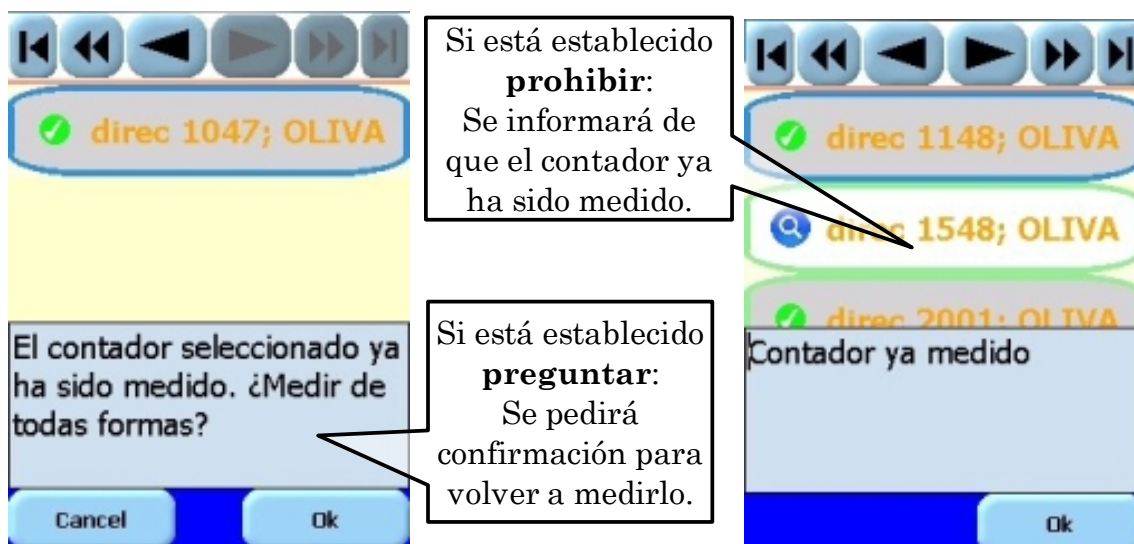
La aplicación previene el abandono no autorizado mediante una contraseña de obligatoria introducción si se desea salir del programa. Ésta contraseña puede cambiarse, pero para ello es necesario conocer la contraseña actual. No se permite una contraseña vacía.



### 1.13 Avisos / Confirmaciones

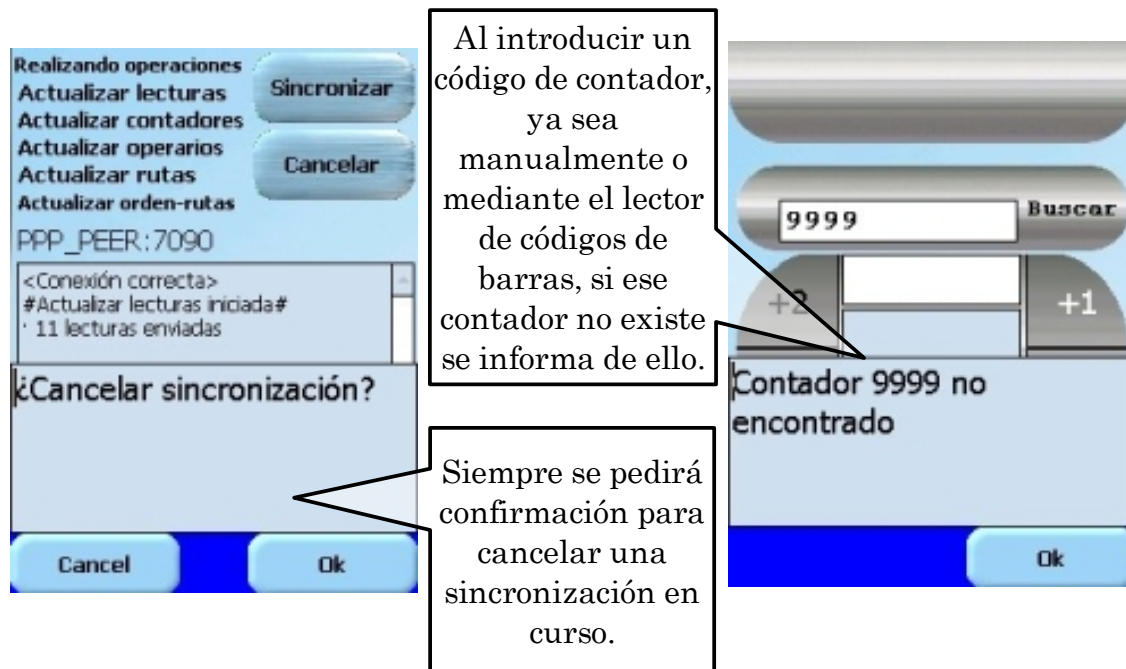
Dependiendo de la configuración establecida en [Configuración general](#), la aplicación puede requerir que el usuario confirme la operación que desea realizar, o que directamente lo prohíba e informe de ello mediante un aviso.

- **Contador ya medido :**

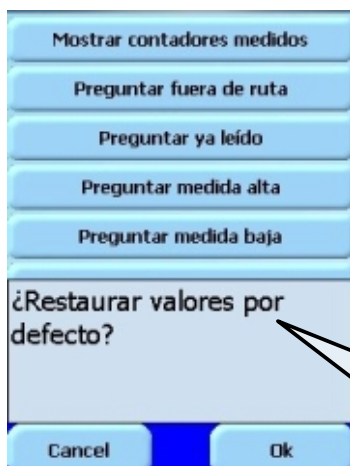


- **Cancelar Sincronización**

- **Contador no encontrado**



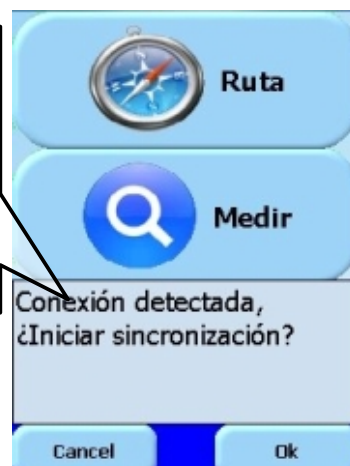
- Valores por defecto



Al detectar la existencia de conectividad la aplicación pedirá confirmación para iniciar una sincronización. (Ver [Sincronización](#))

Siempre se pide confirmación antes de volver a los valores por defecto.

- Iniciar sincronización



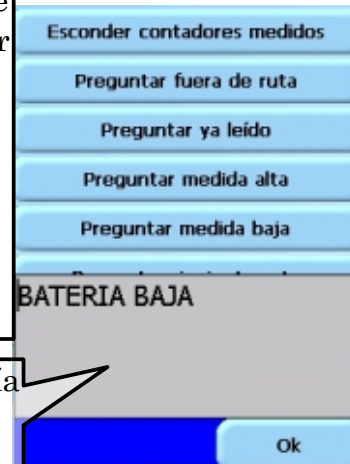
- Medir contador



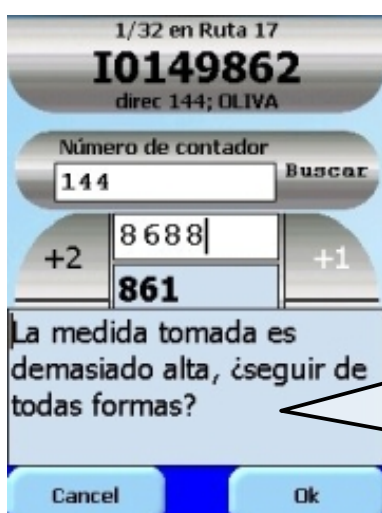
En cualquier momento puede utilizarse el escáner para leer un código de barras con intención de medir un contador. Si esa lectura se hace fuera de la pantalla de toma de lecturas se pedirá confirmación para pasar a medir el contador.

Cuando la carga de la batería se encuentra baja, se mostrará un aviso informando.

- Batería baja

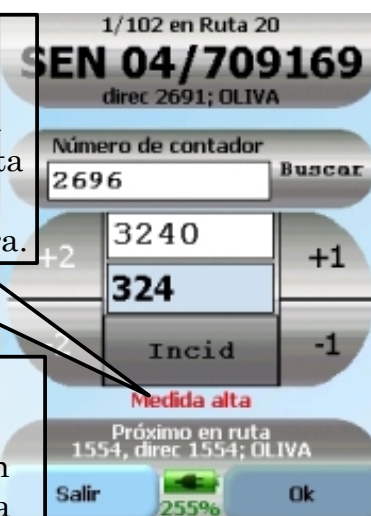


- Medida demasiado alta :



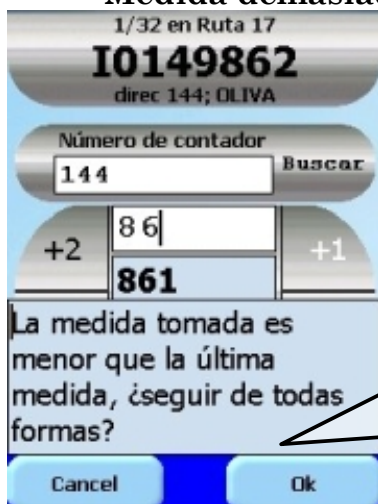
Si está establecido **prohibir:**  
Se informará de que la medida es demasiado alta y se impedirá la introducción de la lectura.

Si está establecido **preguntar:**  
Se pedirá confirmación para aceptar la medida tomada.



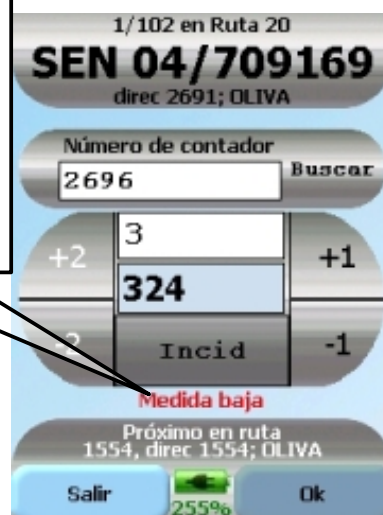


- **Medida demasiado baja :**

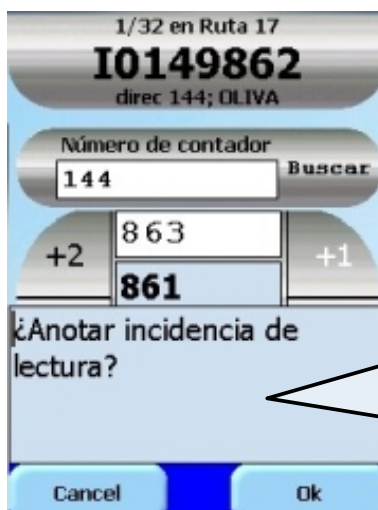


Si está establecido **prohibir:**  
Se informará de que la medida es demasiado baja y se impedirá la introducción de la lectura.

Si está establecido **preguntar:**  
Se pedirá confirmación para aceptar la medida tomada.

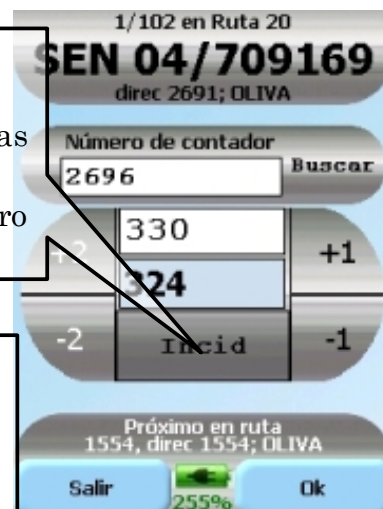


- **Registrar incidencia :**

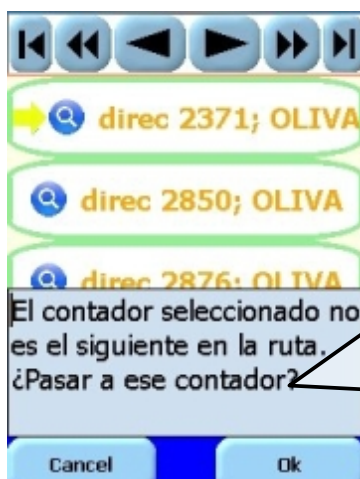


Si está establecido **prohibir:**  
El registro de incidencias estará inhabilitado, impidiendo así el registro de incidencias.

Si está establecido **preguntar:**  
Se pedirá confirmación para registrar la incidencia.

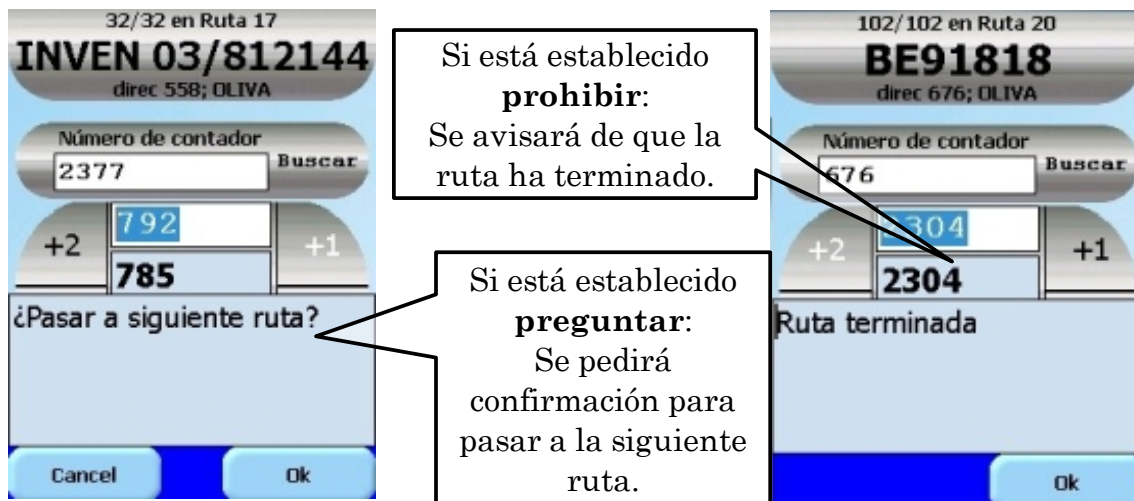


- **Saltar en ruta :**

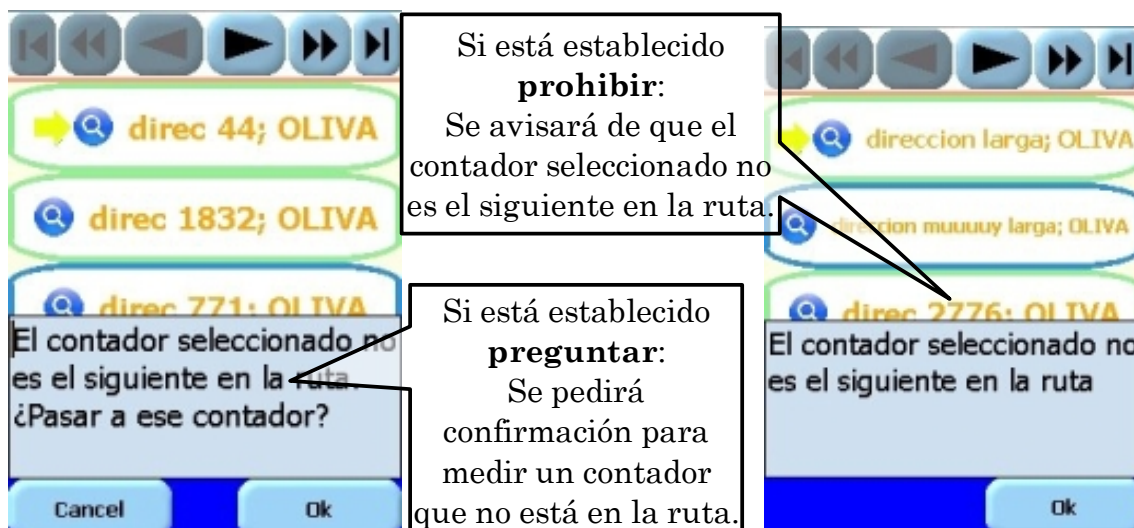


En la pantalla de navegación de ruta, al seleccionar un contador que no es el siguiente a medir se pedirá confirmación para saltar a ese contador.  
Al pulsar **Ok** se pasará a ese contador y se seguirá a partir de él.  
Al pulsar **Cancel** se medirá ese contador pero no se modificará el orden de ruta.

- Pasar a siguiente ruta :



- Medir fuera de ruta :

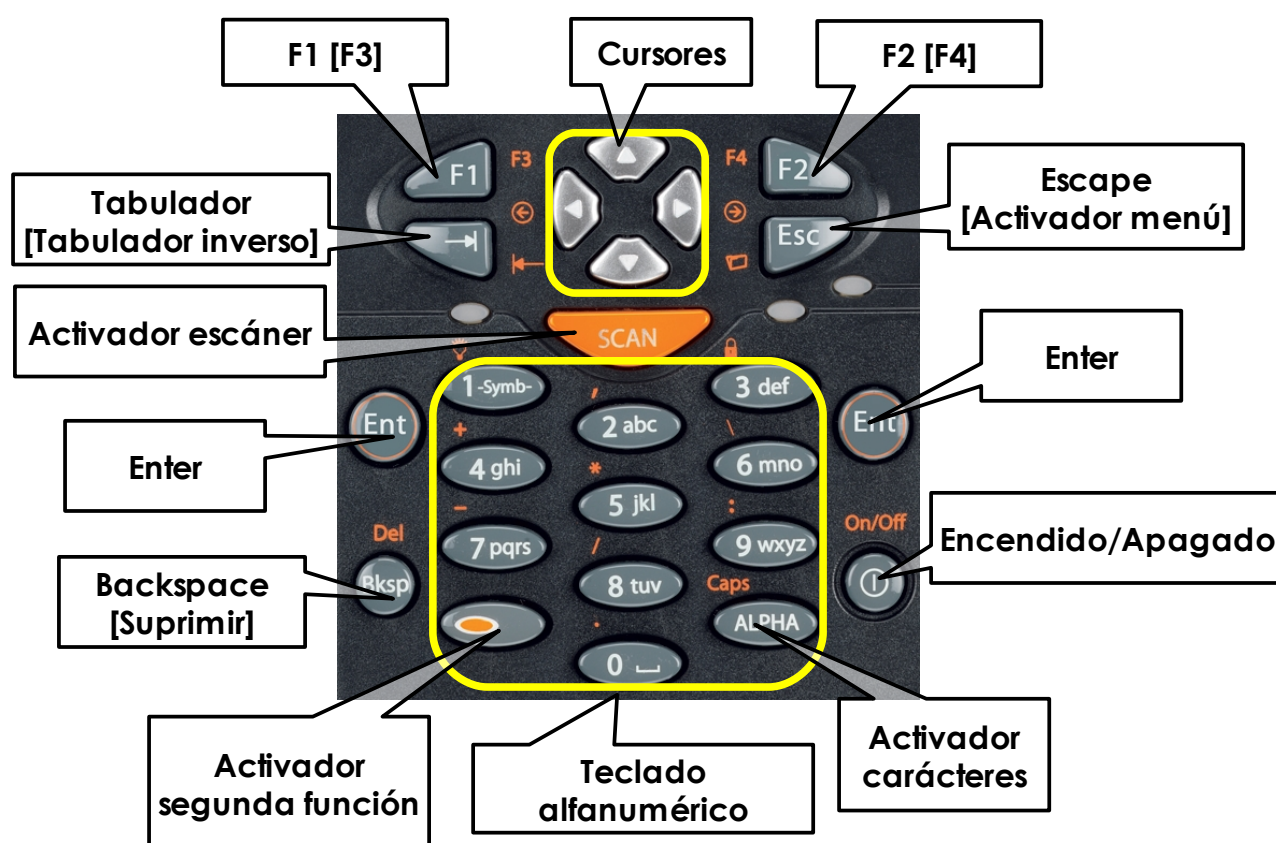


## 2 Adaptación de la aplicación al dispositivo Datalogic Blackjet™

### 2.1 Uso del teclado

Esta aplicación ha sido desarrollada contemplando la posibilidad de su uso en dispositivos móviles que dispongan de teclado. Se muestra el uso del teclado para el dispositivo móvil Datalogic Blackjet™, aunque estas instrucciones son adaptables a cualquier dispositivo móvil con teclado.

#### 2.1.1 Teclado del dispositivo Datalogic Blackjet™



#### 2.1.2 Teclas especiales


- **Tecla naranja, activador de segunda función:**



Esta tecla se utiliza conjuntamente con otra tecla, activando la segunda función (mostrada en naranja encima o al lado de la tecla) de la tecla que se pulsa conjuntamente con ésta.


Por ejemplo: pulsar la tecla **4** mientras se mantiene pulsada la tecla naranja introducirá el carácter **+** en lugar de un **4**.



- **ALPHA**, activador de caracteres: 

Pulsar esta tecla activa la entrada de caracteres de forma que pueden introducirse letras mediante el teclado. Volver a pulsar la tecla desactiva el modo de entrada de caracteres.

Por ejemplo: con el modo de entrada de caracteres activado puede pulsar la tecla **3** una vez para introducir **d**, dos veces para introducir **e** o tres veces para introducir **f**.

- **SCAN**, activador de escáner: 

Pulse y mantenga pulsada esta tecla para realizar una lectura de un código de barras mediante el escáner.



- **BACKSPACE**, borrar : 

Pulse esta tecla para borrar caracteres de los [campos de entrada de datos](#).

## 2.2 Notas de navegación

Si el dispositivo móvil dispone de teclado, éste puede utilizarse para navegar por la interfaz de la aplicación.

### 2.2.1 En todas las pantallas

- Teclas **F1** y **F2**:  

En todas las pantallas existen uno o dos botones abajo que normalmente se corresponden con las opciones de **Aceptar** ó **Cancelar**, **Salir** ó **Entrar**, **Salir** ó **Ok**, etc.



- En estas pantallas, pulsar la tecla **F1** equivale a pulsar el botón de abajo a la izquierda, y pulsar **F2** equivale a pulsar el botón de abajo a la derecha.
- En caso de que sólo haya un botón, éste se activa con la tecla **F1** si el botón se encuentra a la izquierda o con **F2** si se encuentra a la derecha.

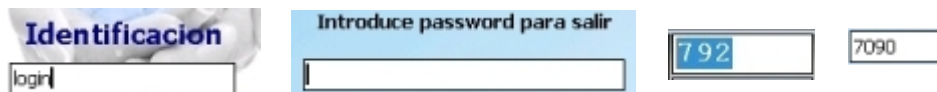
- Tecla **Escape**: 



En todas las pantallas la tecla **Escape** abandona la pantalla, cancelando los cambios que se hayan realizado.

- Pantalla de bloqueo:  
La pantalla de bloqueo es especial, y tiene reglas especiales para poder abandonarse.

### 2.2.2 Campos de entrada de datos

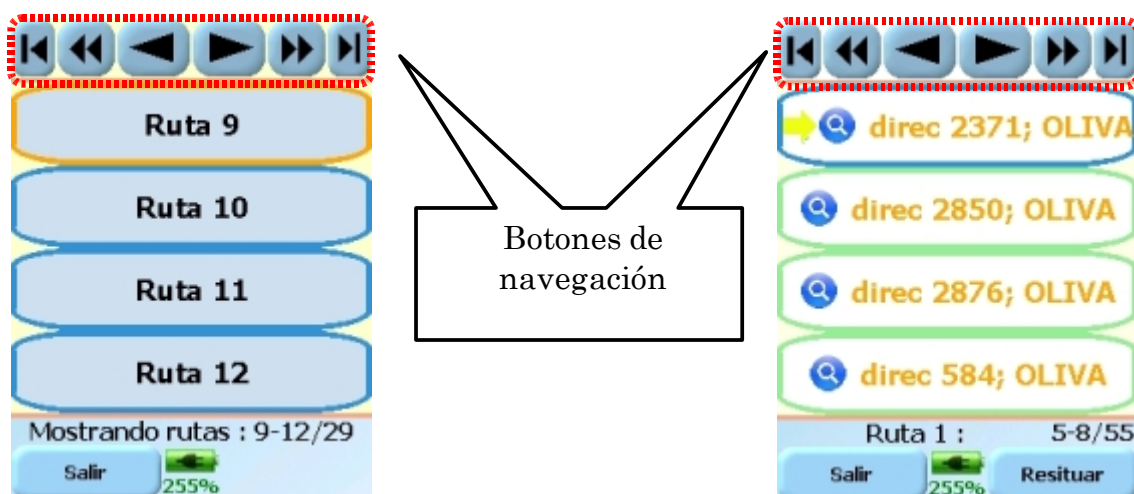
- Ejemplos de campos de entrada de datos:




- Entrada de datos con teclado del dispositivo:  
En estos campos los datos pueden introducirse utilizando el teclado en pantalla o el teclado del dispositivo en caso de que disponga de uno ( Ver [Teclado del dispositivo](#) ).
- Tecla **Enter**:   
Pulsar la tecla **Enter** en estos campos equivale a pulsar el botón de aceptación de la pantalla en la que se encuentra el campo.
- Selección de datos:  
En estos campos puede realizarse una selección de todos los datos al hacer doble clic sobre el campo, o realizar una selección parcial arrastrando el puntero sobre los datos a seleccionar. Introducir datos mientras se tienen datos seleccionados hace que los nuevos datos sustituyan los datos seleccionados. Por ejemplo: selección de todos los datos : 
- Campos que sólo aceptan dígitos:  
Algunos de estos campos de entrada de datos aceptan únicamente la entrada de dígitos. Estos campos no aceptan la introducción de caracteres.

### 2.2.3 Cursores para navegación en rutas







- Pantallas de [Selección de Ruta](#) y de [Navegación en Ruta](#):




- Botones de navegación: 

Los botones de navegación permiten avanzar o retroceder entre todas las rutas o todos los contadores disponibles cambiando la selección

que se muestra en pantalla. Existen seis botones de navegación con este comportamiento:

-  Retrocede hasta el inicio, mostrando las primeras rutas o los primeros contadores existentes.
-  Avanza hasta el final, mostrando la última ruta o el último contador existente.
-  Retrocede la mitad de elementos existentes, esto es, si existen 100 elementos retrocederá 50 elementos.
-  Avanza la mitad de elementos existentes, esto es, si existen 100 elementos avanzará 50 elementos.
-  Retrocede un paso, mostrando las rutas o contadores inmediatamente anteriores a los que se están mostrando.
-  Avanza un paso, mostrando las rutas o contadores inmediatamente posteriores a los que se están mostrando.

- Uso de los cursores:   
Si el dispositivo móvil dispone de cursores, éstos pueden emplearse para navegar por las pantallas de [Selección de Ruta](#) y [Navegación en Ruta](#). Los cursores izquierdo y derecho hacen retroceder o avanzar un paso los elementos mostrados respectivamente. Los cursores arriba y abajo permiten cambiar el elemento seleccionado. El elemento seleccionado está resaltado con un color de borde diferente al resto. Por ejemplo:



- Pulsar la tecla **Enter** pasa a seguir la ruta seleccionada o a medir el contador seleccionado.
- Ver [Selección de ruta](#) y [Seguimiento de ruta](#).

## 3 Instalación de la aplicación

### 3.1 Requisitos

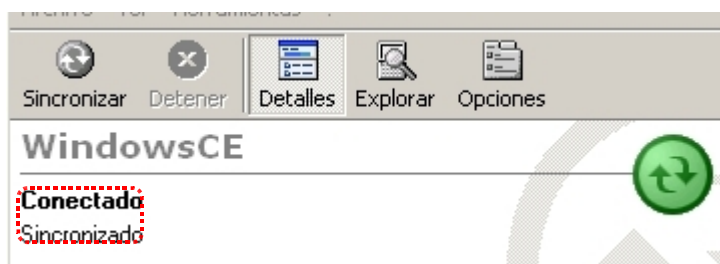
Para que la aplicación pueda funcionar, el dispositivo destino debe tener instalados estos servicios/programas:

- *.Net Compact Framework 2.0.*
- *SQL Server Mobile* ( también conocido como *SQL Server Everywhere* ).
- *SQL Server Mobile Client.*
- Vea [Instalación de requisitos](#).

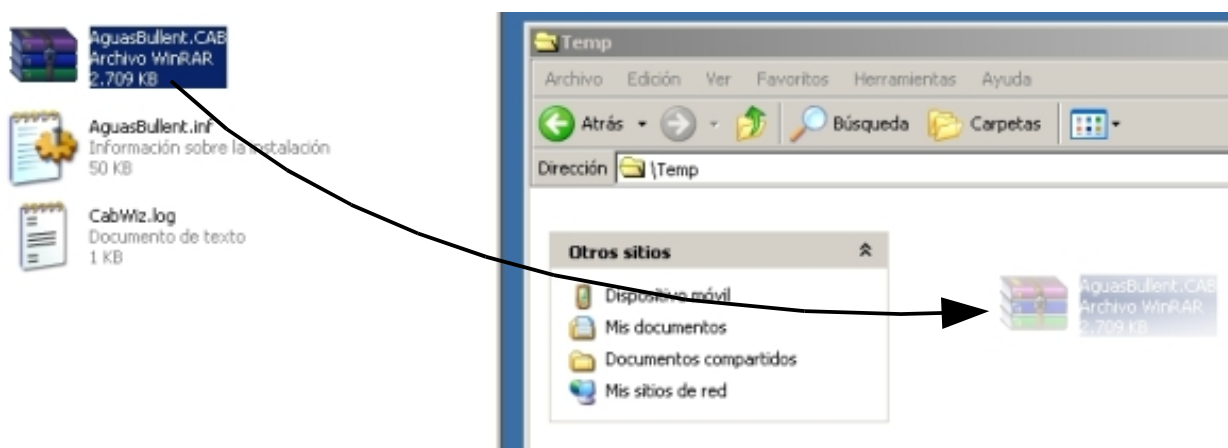
### 3.2 Instalación en dispositivo Datalogic BlackJet™

Estos pasos están dirigidos a la instalación de la aplicación en el dispositivo Datalogic Blackjet™. Si va a realizar la instalación en otro dispositivo vea [Instalación general](#). Esta instalación asegura que la aplicación queda instalada de forma permanente en el dispositivo.

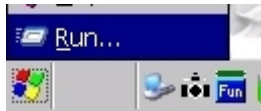
- 1) Conecte el dispositivo al ordenador mediante *ActiveSync*. Espere a que *ActiveSync* termine la sincronización.



- 2) Copie el archivo **AguasBullent.CAB** a la carpeta **\temp** del dispositivo. Cree la carpeta **\temp** si no existe en el dispositivo.



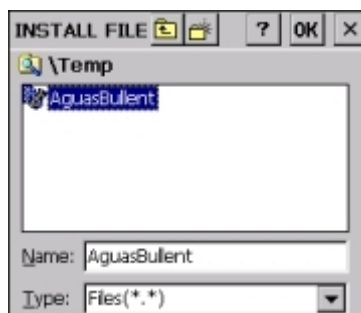
- 3) En el dispositivo, haga clic en **Inicio (Start) -> Ejecutar... (Run...)**.



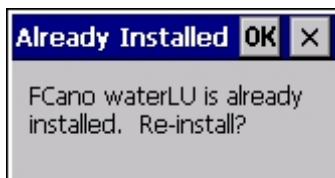
- 4) Seleccione el ejecutable **\Windows\safe\_setup** y haga clic en **Ok**.



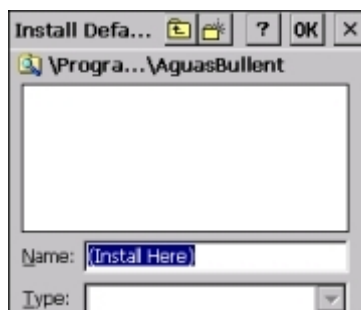
- 5) En el nuevo cuadro de diálogo seleccione el archivo copiado previamente, **\temp\AguasBullent.CAB** y haga clic en **Ok**.



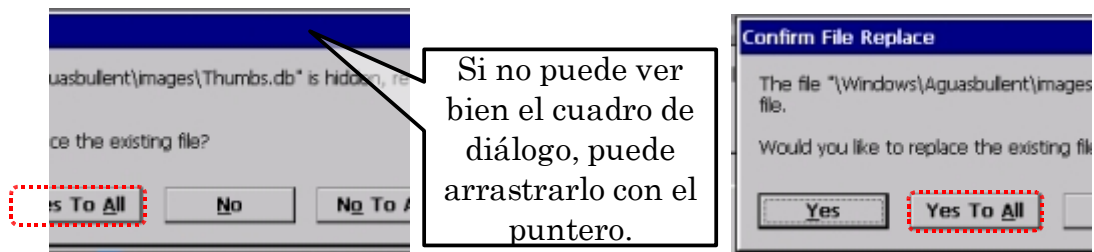
\* Si **Windows** le informa de que el programa ya está instalado y le pregunta si desea reinstalarlo haga clic en **Ok**.



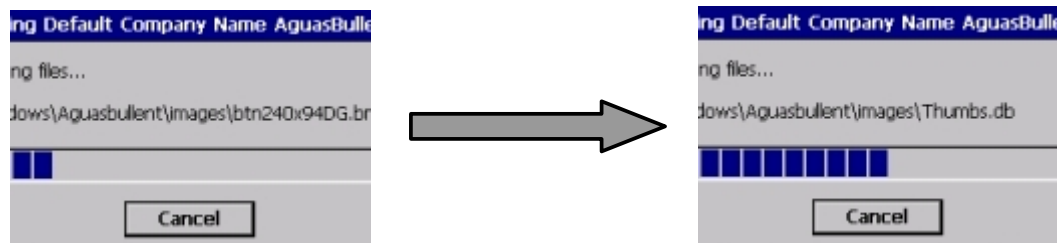
- 6) Elija la ruta de instalación y haga clic en **Ok**. Puede dejar la opción por defecto.



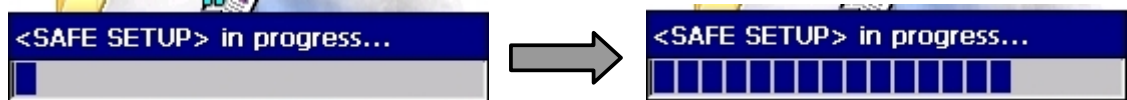
\* Si **Windows** le pide confirmación para sobrescribir los ficheros haga clic en **Sí a todo (Yes to all)**.



- 7) Verá el progreso del proceso de instalación.



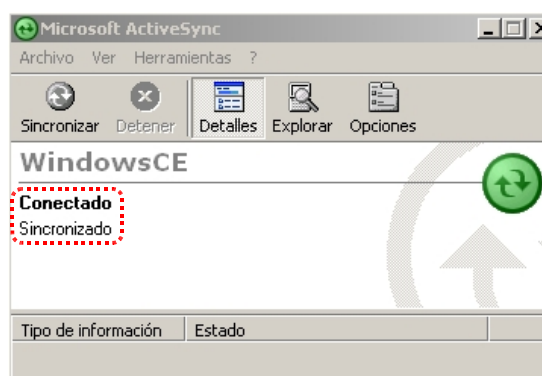
- 8) Una vez concluida la instalación, se realizará una copia en la memoria no volátil para hacer permanente la instalación.



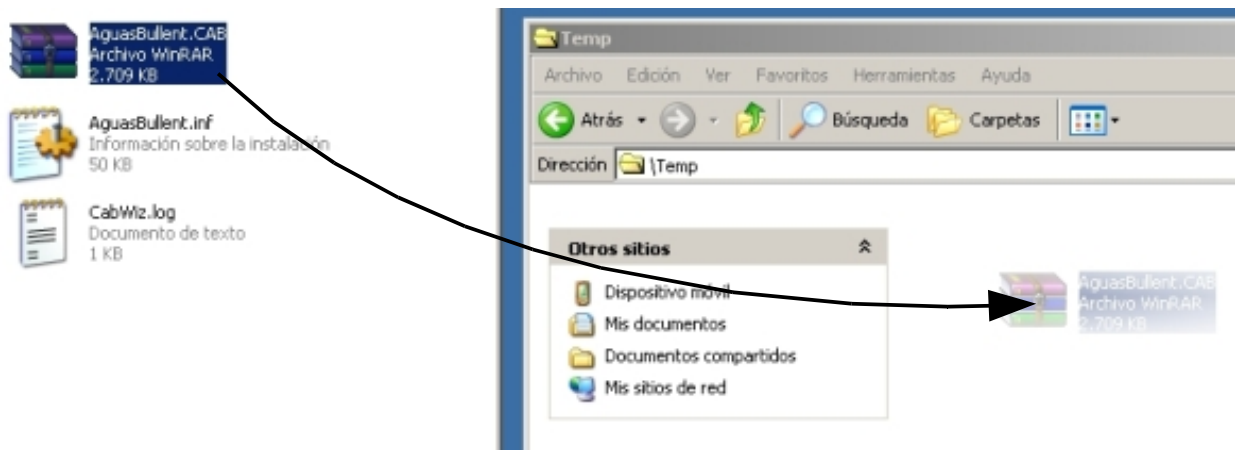
- 9) La aplicación ya está instalada.

### 3.3 Instalación general

- 1) Conecte el dispositivo al ordenador mediante *ActiveSync*. Espere a que *ActiveSync* termine la sincronización.



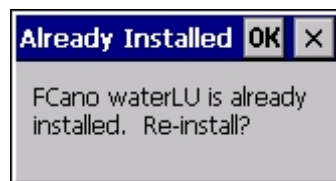
- 2) Copie el archivo **AguasBullent.CAB** a la carpeta **\temp** del dispositivo. Cree la carpeta **\temp** si no existe en el dispositivo.



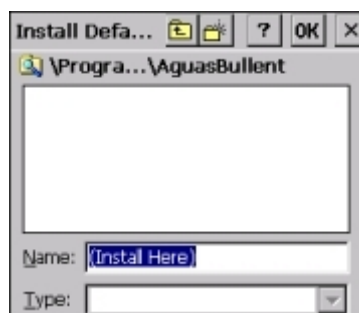
- 3) En el dispositivo, abra un explorador de archivos y navegue hasta la carpeta **temp**. Ejecute el archivo copiado **AguasBullent.CAB**.



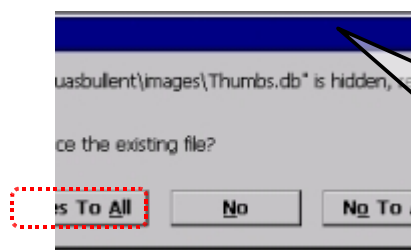
\* Si **Windows** le informa de que el programa ya está instalado y le pregunta si desea reinstalarlo haga clic en **Ok**.



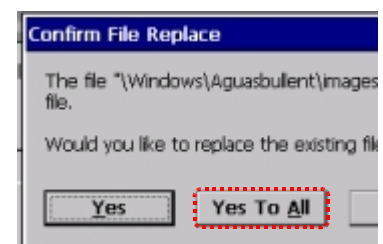
- 4) Elija la ruta de instalación y haga clic en **Ok**. Puede dejar la opción por defecto.



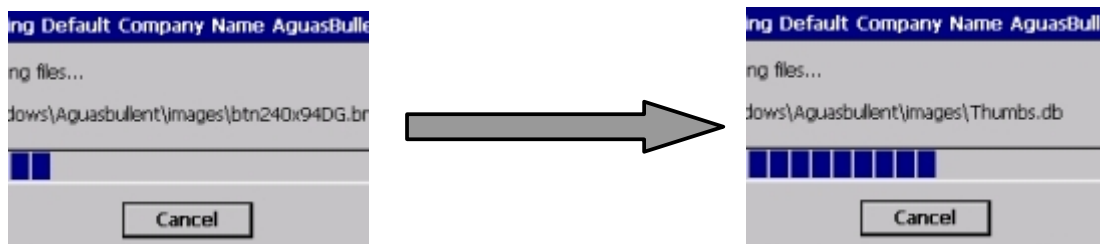
\* Si **Windows** le pide confirmación para sobrescribir los ficheros haga clic en **Sí a todo (Yes to all)**.



Si no puede ver bien el cuadro de diálogo, puede arrastrarlo con el puntero.



5) Verá el progreso del proceso de instalación.



6) La aplicación ya está instalada.

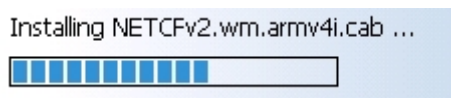
### 3.4 Instalación de requisitos

Se proporcionan los archivos necesarios para instalar en el dispositivo los programas/servicios necesarios por la aplicación. La instalación de estos requisitos debe realizarse en el orden mostrado.

Puede existir más de un instalador para un requisito. Debe instalar el adecuado para su dispositivo. Vea [Identificación del instalador](#).

- *.Net Compact Framework 2.0*

Copie el archivo de instalación en la carpeta **\temp** del dispositivo. Ejecútelo en el dispositivo y se instalará *.Net Compact Framework 2.0*. Verá una barra de progreso mientras dure la instalación.



Puede que tenga que reiniciar el dispositivo después de la instalación.

\* La mayoría de dispositivos con un sistema **Windows** tienen preinstalado *.Net Compact Framework 2.0* de fábrica.

- *SQL Server Mobile ( SQL Server Everywhere )*

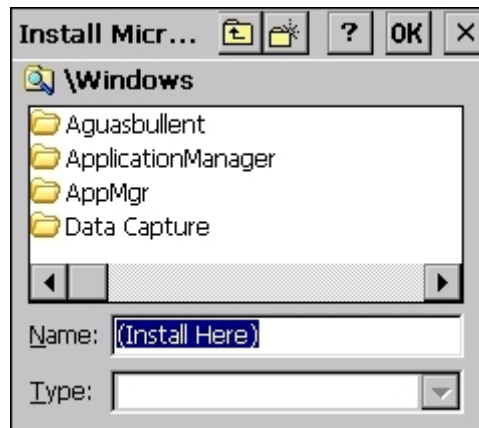
Copie el archivo de instalación en la carpeta **\temp** del dispositivo. Ejecútelo en el dispositivo.

\* Si **Windows** le informa de que el programa ya está instalado y le pregunta si desea reinstalarlo haga clic en **Ok**.



\* Si **Windows** le pregunta por la ruta de instalación, es importante que deje la ruta por defecto.





Se mostrará una barra de progreso. Cuando termine, *SQL Server Mobile* ( *SQL Server Everywhere* ) estará instalado en el dispositivo.

- *SQL Server Mobile Client*  
 Copie el archivo de instalación en la carpeta **\temp** del dispositivo.  
 Ejecútelo en el dispositivo.  
 \* Si **Windows** le informa de que el programa ya está instalado y le pregunta si desea reinstalarlo haga clic en **Ok**.



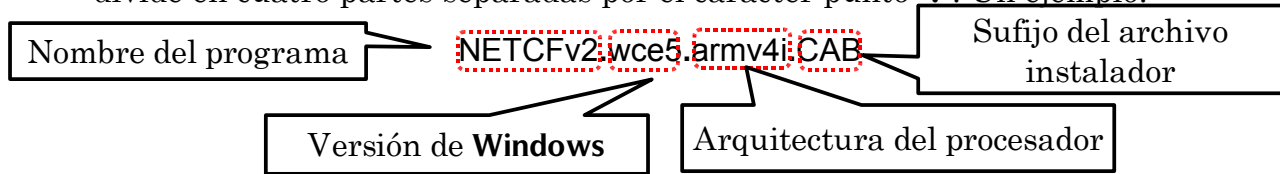
\*Si **Windows** le pregunta por la ruta de instalación, **es importante** que deje la ruta por defecto.



Se mostrará una barra de progreso. Cuando termine, *SQL Server Mobile Client* estará instalado en el dispositivo.

### 3.4.1 Identificación del instalador para un dispositivo y sistema

El nombre de los archivos instaladores para dispositivos sigue un formato que permite identificar cuál de ellos es el correcto para un dispositivo y sistema. Por norma general, el nombre de un instalador se divide en cuatro partes separadas por el carácter punto “.”. Un ejemplo:



Los nombres de programa para los requisitos son :

- Para *.Net Compact Framework 2.0* : NETCFv2
- Para *SQL Server Mobile ( SQL Server Everywhere )* : sqlce30
- Para *SQL Server Mobile Client* : sql

Las versiones de **Windows** típicas son : **Windows** CE 4.0 (wce4), **Windows** CE 5.0 (wce5), **Windows** CE 6.0 (wce6), **Windows** Mobile (wm).

Las versiones de arquitectura del procesador típicas son : ARM (armv4i), MPIS 2 (mipsii), MIPS 4 (mipsiv), x86 (x86).

El dispositivo Datalogic Blackjet™ tiene un procesador ARM (armv4i) y emplea **Windows** CE 5.0 (wce5), por lo que un instalador para este dispositivo tiene esta forma: <nombre>.wce5.armv4i.cab

## 4 Tras la instalación

Después de haber realizado una instalación de la aplicación en un dispositivo es recomendable llevar a cabo estas tareas de forma que la aplicación quede totalmente preparada para llevar a cabo tareas de lecturas y seguimiento de rutas.

\* Tras una instalación se crea un operario temporal para poder identificarse en la aplicación (Vea [Identificación](#)) y realizar la primera sincronización. Éste operario es “987”. Se borrará tras la primera sincronización. La única forma de entrar en el sistema tras una instalación es empleando este operario.

- **Cambie la contraseña de abandono de la aplicación.** Tras una instalación la contraseña de abandono de la aplicación es “147”. Se recomienda cambiar esa contraseña por otra secreta.
- **Especifique las opciones que prefiera para la configuración del dispositivo** en la aplicación cliente. Consulte el apartado Configuración de los dispositivos desde [AGUASCLI](#) para ver cómo hacerlo.
- **Realice una sincronización completa**, con todas las operaciones (consulte el apartado [Sincronización](#).) La base de datos de la aplicación está vacía tras una instalación. Con esta sincronización queda preparada con datos actuales. Esta sincronización también establece la configuración inicial desde el servidor al dispositivo. Si no se realiza una sincronización el dispositivo no dispondrá de datos con los que trabajar.

## 5 Importante

Aún con la batería plenamente cargada y colocada en el dispositivo, si el dispositivo se deja un largo período de tiempo sin cargar la batería se descargará y se podrían perderse algunos datos. Es importante que antes de dejar el dispositivo sin emplear durante un largo período de tiempo realice una sincronización con la operación *Actualizar Lecturas* (consulte el apartado [Sincronización](#)) para asegurarse que no pierde ningún dato que no pueda recuperar.

Dejar los dispositivos Datalogic Blackjet™ sin batería durante largos períodos de tiempo (2 o más días) no hará que se borre la aplicación del dispositivo, siempre que se haya realizado la instalación siguiendo los pasos que se indican en este manual. No obstante, la aplicación volverá a un estado idéntico al que se encuentra tras una instalación. Es recomendable realizar los pasos descritos en el apartado [Tras la instalación](#) al volver a emplear el dispositivo tras un largo período de tiempo sin batería.

Antes de dejar los dispositivos durante largos períodos de tiempo (2 o más días) sin emplear, es obligatorio realizar una sincronización con al menos la operación *Actualizar Lecturas* (consulte el apartado [Sincronización](#)), de otra forma las lecturas que aún estén pendientes de ser descargadas al servidor central podrían perderse.

Otro tipo de dispositivo podría disponer de mecanismos que permitan mantener la aplicación en memoria aún sin batería. Consulte el manual del dispositivo para comprobarlo.

Si la aplicación se instala sin emplear ningún mecanismo que garantice su permanencia en memoria aún cuando el dispositivo no tenga batería, al retirar la batería del dispositivo durante largos períodos de tiempo hará que se borre la aplicación del dispositivo y los datos, incluyendo cualquier lectura o incidencia no actualizada.

En el dispositivo Datalogic Blackjet™ no se perderá la aplicación si se siguieron las instrucciones de instalación (consulte [Instalación en dispositivo Datalogic Blackjet™](#)), pero sí se perderán los datos tras un largo período sin batería, por lo que es obligatorio realizar una sincronización con la operación *Actualizar Lecturas* (consulte el apartado [Sincronización](#)) antes de retirar la batería del dispositivo un largo tiempo.

- Página intencionadamente en blanco -





**10.2.- Manual de uso de nuevas funcionalidades de  
AguasCli**

Manual de uso de sincronización e impresión  
de códigos de barras en **AguasCli**





- Página intencionadamente en blanco -

# Índice

I - Proceso de sincronización en ordenador de escritorio.....	5
II - Configuración de los dispositivos desde AguasCli.....	10
II.1 - Configuración avanzada.....	13
III - Impresión de códigos de barras.....	14
IV - Impresión en Zebra.....	16
IV.1 - Instalación de la impresora Zebra.....	16
IV.1.a - Instalación de Zebra S4M.....	17
IV.1.b - Instalación de ZebraDesigner.....	21
IV.1.c - Instalación de impresora ZDesigner.....	22
IV.1.d - Configuración de ZDesigner.....	24

Este manual asume:

- que el usuario está familiarizado con el uso de la aplicación **AGUASCLI**.
- que el usuario sabe como conectar el dispositivo a un equipo mediante *ActiveSync*, o bien activar las capacidades WiFi o similar que permitan acceder a la red.
- que se dispone de la última versión de **AGUASCLI**.
- que el usuario tiene acceso tanto al manual de uso e instalación de **WATERLU**, como al menos a un dispositivo con el sistema **WATERLU** instalado y funcionando.

Este manual emplea estos términos/generalidades:

- **dispositivo** o **dispositivo móvil**: cualquier dispositivo, ya sea una PDA, un SmartDevice, PocketPC, etc. sobre el que puede ejecutarse **Windows** CE 5.0, y por ende, esta aplicación.
- **lectura** o **medida** : el hecho de tomar la lectura de consumo de un contador o el propio valor de consumo de un contador.
- **AGUASCLI** : programa que es la parte cliente del sistema generado con **OLIVANOVA**. Es la aplicación que se ejecuta sobre un computador de sobremesa.
- **servidor** o **servidor central** : programa que es la parte servidora del sistema generado con **OLIVANOVA**. Es la aplicación que se ejecuta en el servidor.
- **conectividad** o **disponer de conexión** : se produce cuando, o bien el dispositivo está conectado a un equipo mediante *ActiveSync*, o bien está conectado a una red mediante WiFi, Bluetooth, etc.

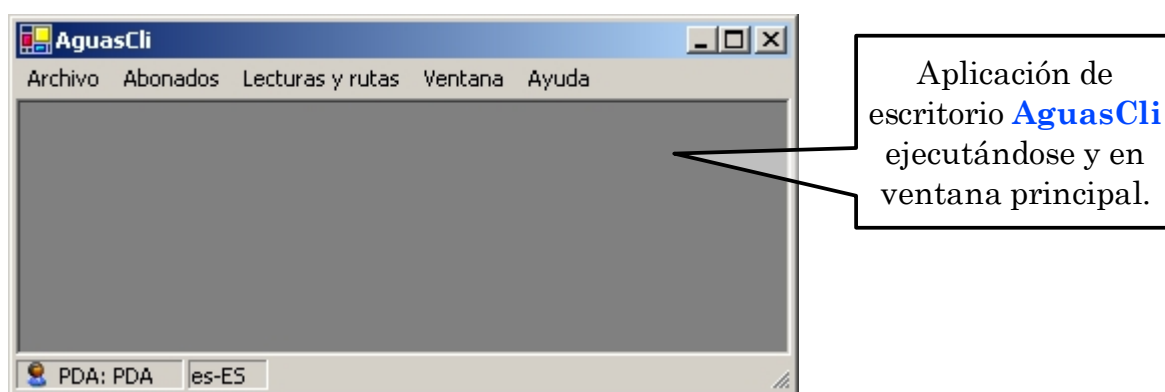
# 1 Proceso de sincronización en ordenador de escritorio

La aplicación de escritorio actúa de servidor intermedio entre el dispositivo y el servidor central cuando se realiza una sincronización. Es por ello que la aplicación de escritorio debe estar ejecutándose para llevar a cabo la sincronización.

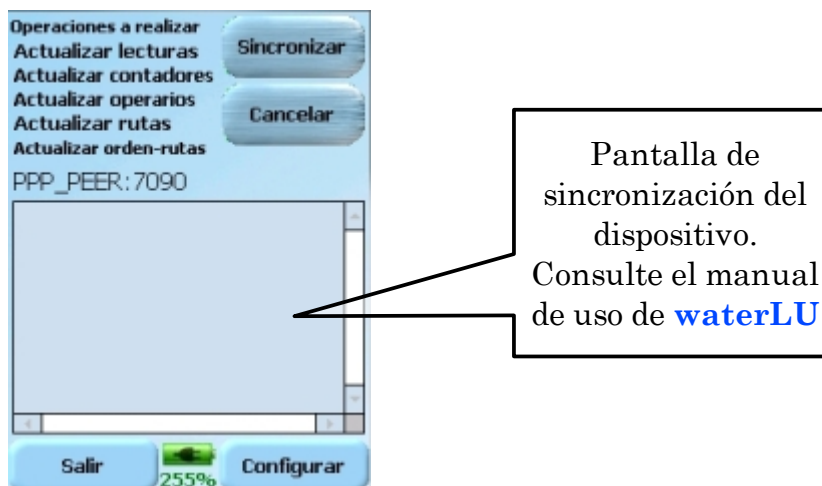
Los pasos a seguir para iniciar una sincronización se pueden ver en el apartado Sincronización del manual de [WATERLU](#).

La buena ejecución de la sincronización requiere que en la aplicación de escritorio se lleven a cabo algunos pasos, descritos aquí.

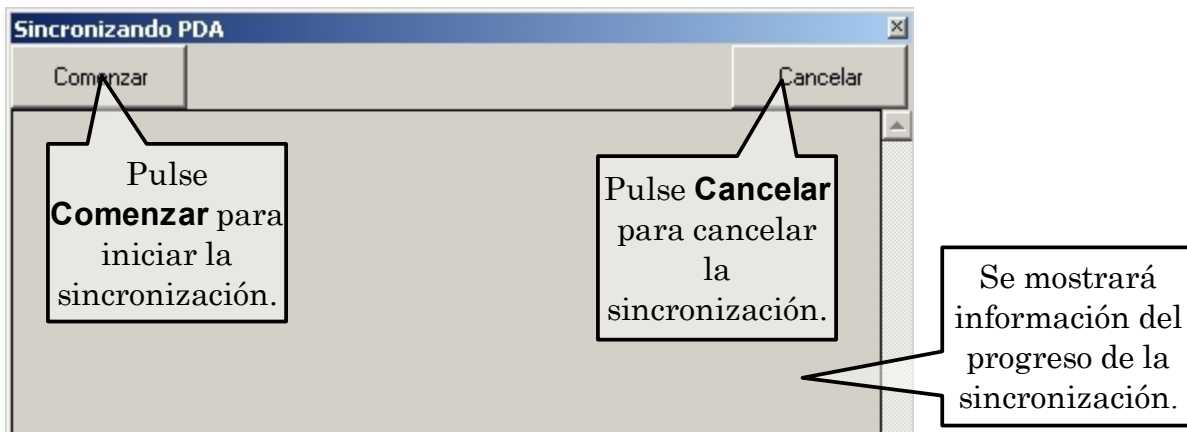
## 1) Inicio de la aplicación de escritorio



## 2) Inicio del proceso de sincronización desde el dispositivo. Para más información, consulte el manual de uso de la aplicación [WATERLU](#)



## 3) Confirmación de inicio de sincronización.



- 4) El dispositivo y la aplicación de escritorio se conectan.

```
<Conexión correcta>
```

- 5) Se realizan las operaciones seleccionadas en el dispositivo secuencialmente. En este orden :  
*Actualizar lecturas, Actualizar contadores, Actualizar operarios, Actualizar rutas, Actualizar orden-rutas, Actualizar Configuración.*

- 6) **Actualizar lecturas:** (siempre que haya lecturas pendientes)  
 Pregunta por la fecha de las lecturas. Esta fecha corresponde con el trimestre de facturación de las lecturas.



Elija la fecha a la que corresponden las lecturas.



Descarga de las lecturas e incidencias al servidor.

```
#Actualizar lecturas iniciada#
• 38 lecturas recibidas
• 38 lecturas actualizadas
#Actualizar lecturas terminada#
```

Al terminar se muestra informe de las lecturas e incidencias descargadas.

Las incidencias y lecturas se muestran agrupadas por categorías. Para cada categoría se muestra la descripción general de la incidencia y el número de casos.

Resumen actualización lecturas	
+	Incidencias de tipo 7 (Otro) : 3
+	Incidencias de tipo 6 (Contador obstruido) : 3
+	Incidencias de tipo 5 (Rotura accesorios contador) : 2
+	Incidencias de tipo 4 (Desplazar contador) : 3
+	Incidencias de tipo 3 (Rotura contador) : 3
+	Incidencias de tipo 2 (Medida baja) : 4
+	Incidencias de tipo 1 (Medida alta) : 6
+	Incidencias de tipo 0 (Todo correcto) : 14

Informe de las lecturas e incidencias descargadas desde el dispositivo.

Las incidencias detalladas muestran el código de contador e información detallada de la incidencia.

Resumen actualización lecturas	
-	Incidencias de tipo 7 (Otro) : 3
---	580 - Otro
---	1449 - Otro
---	2596 - Otro
-	Incidencias de tipo 6 (Contador obstruido) : 3
---	598 - Contador obstruido
---	599 - Contador obstruido
---	704 - Contador obstruido
-	Incidencias de tipo 5 (Rotura accesorios contador) : 2
---	591 - Rotura accesorios contador
---	595 - Rotura accesorios contador
-	Incidencias de tipo 4 (Desplazar contador) : 3
---	585 - Desplazar contador
---	593 - Desplazar contador
---	2800 - Desplazar contador
-	Incidencias de tipo 3 (Rotura contador) : 3
---	586 - Rotura contador
---	587 - Rotura contador
---	588 - Rotura contador
-	Incidencias de tipo 2 (Medida baja) : 4
---	361 - Lectura baja, medido: 31, anterior: 3184
---	363 - Lectura baja, medido: 36, anterior: 3640
---	759 - Lectura baja, medido: 21, anterior: 2124
---	890 - Lectura baja, medido: 13, anterior: 1320
-	Incidencias de tipo 1 (Medida alta) : 6
---	360 - Lectura alta, medido: 28910, esperado: 28910
---	365 - Lectura alta, medido: 24220, esperado: 24220
---	579 - Lectura alta, medido: 22780, esperado: 22780
---	589 - Lectura alta, medido: 19100, esperado: 19100
---	590 - Lectura alta, medido: 10870, esperado: 10870
---	592 - Lectura alta, medido: 22900, esperado: 22900
-	Incidencias de tipo 0 (Todo correcto) : 14
---	362 - lectura correcta
---	367 - lectura correcta
---	578 - lectura correcta
---	584 - lectura correcta
---	673 - lectura correcta

Otras incidencias.

Incidencias de contador obstruido.

Incidencias por rotura de accesorios de contador.

Incidencias de desplazar contador.

Incidencias por rotura de contador.

Incidencias por lectura baja: muestra la lectura real y la lectura anterior.

Incidencias por lectura alta: muestra la lectura real y la lectura esperada

Lecturas realizadas correctamente, sin incidencia.

## 7) Actualizar contadores:

Descarga la información de contadores del dispositivo.  
Compara con la información del servidor central.  
Envía los contadores a añadir, a eliminar y los que han cambiado al dispositivo.

```
#Actualizar contadores iniciada#
• 0 contadores recibidos
• Contadores comparados
• 0 contadores a borrar enviados
• 0 contadores a actualizar enviados
• 3008 contadores a añadir enviados
#Actualizar contadores terminada#
```

#### 8) Actualizar operarios:

Descarga la información de operarios del dispositivo.  
Compara con la información del servidor central.  
Envía los operarios a añadir, a eliminar y los que han cambiado al dispositivo.

```
#Actualizar operarios iniciada#
• 1 operarios recibidos
• Operarios comparados
• 1 operarios a borrar enviados
• 0 operarios a actualizar enviados
• 3 operarios a añadir enviados
#Actualizar operarios terminada#
```

#### 9) Actualizar rutas:

Descarga la información de rutas del dispositivo.  
Compara con la información del servidor central.  
Envía rutas a añadir, a eliminar y las que han cambiado al dispositivo.

```
#Actualizar rutas iniciada#
• 29 rutas recibidas
• Rutas comparadas
• 0 rutas a borrar enviadas
• 0 rutas a actualizar enviadas
• 0 rutas a añadir enviadas
#Actualizar rutas terminada#
```

#### 10) Actualizar orden-Rutas:

Descarga la información de orden de los contadores en las rutas del dispositivo.  
Compara con la información del servidor central.  
Envía los nuevos contadores en rutas, los contadores se han borrado de su ruta y los contadores que han cambiado de orden en su ruta al dispositivo.

```
#Actualizar orden-rutas iniciada#
• 0 ordenRutas recibidas
• OrdenRutas comparadas
• 0 ordenRutas a borrar enviadas
• 0 ordenRutas a actualizar enviadas
• 2867 ordenRutas a añadir enviadas
#Actualizar orden-rutas terminada#
```

#### 11) Actualizar configuración :

Tras la actualización de datos, se establece la configuración del dispositivo con los valores que indica el servidor. Consulte el apartado [Configuración de los dispositivos desde AGUASCLI](#).

```
#Actualizar configuracion iniciada#  
#Actualizar configuracion terminada#
```

## 12) Desconexión del dispositivo:

```
<Desconexión correcta>
```

Tras la desconexión, el dispositivo no necesita interactuar más con la aplicación de escritorio ni el servidor central. Puede cerrar la ventana de sincronización y seguir con las tareas habituales.

## 13) Al terminar todo el proceso se muestra información del tiempo invertido:

```
#duración: 0m:59s.264ms#
```

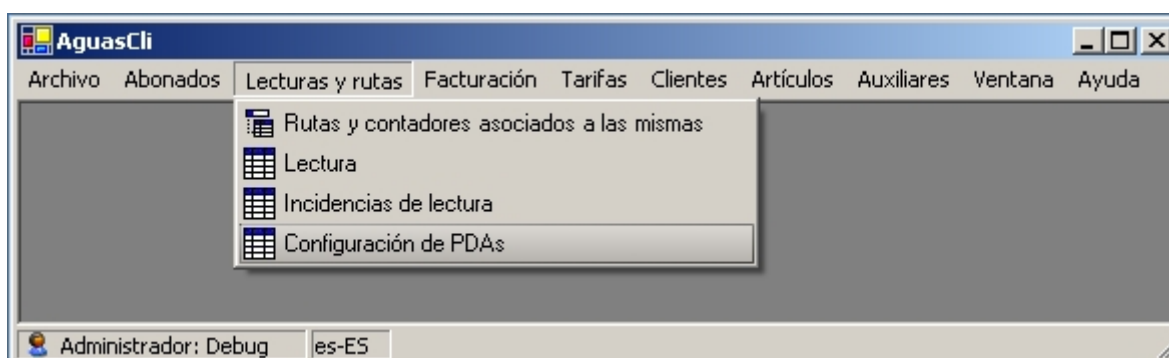
Durante el proceso, la aplicación de escritorio **no** podrá emplearse para realizar otras tareas. La duración total del proceso dependerá de las operaciones a realizar y de la cantidad de información que deba procesarse. El proceso puede llegar a durar varios minutos.



## 2 Configuración de los dispositivos desde AGUASCLI

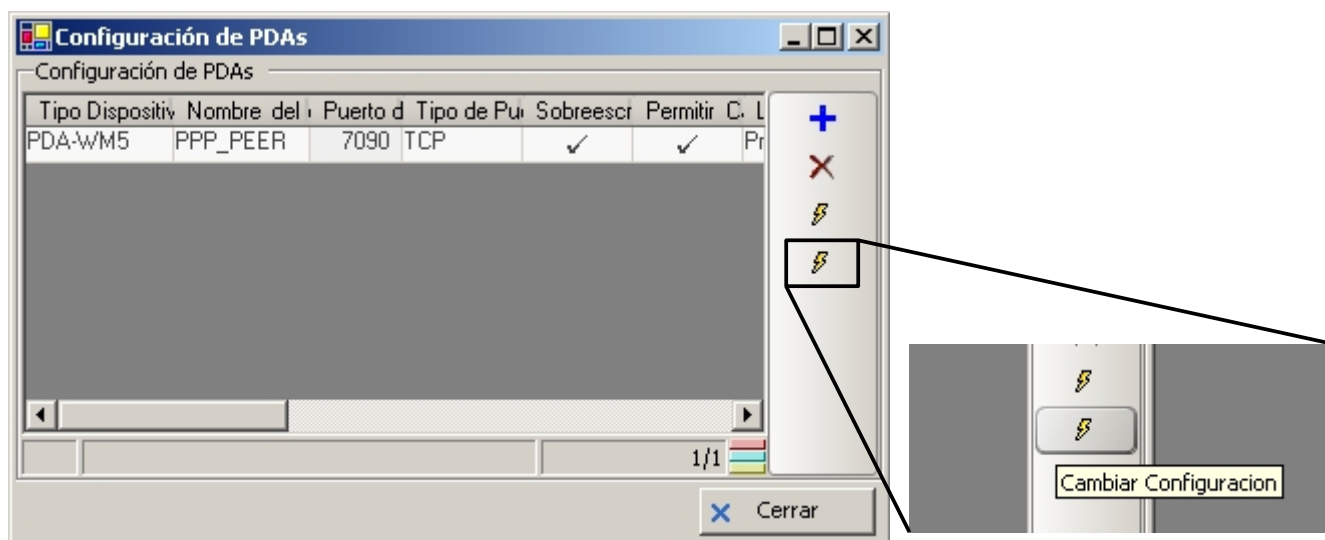
Se da la opción de configurar los dispositivos móviles desde la aplicación **AGUASCLI**. Los perfiles que tienen acceso a la configuración del dispositivo son **Administrador** y **PDA**.

La configuración establecida no se hará efectiva en los dispositivos hasta que estos no se sincronicen correctamente, pues es en el proceso de sincronización cuando se transfiere la configuración establecida aquí a los dispositivos, y únicamente si la opción **Sobreescribir Configuración PDA** está marcada.



Para acceder a la configuración de PDA se ha añadido al menú de **Lecturas y rutas** la opción de **Configuración de PDAs**.

La ventana que se abre contiene un listado de las configuraciones para dispositivos existentes. Puede existir una configuración para cada tipo de dispositivo de que se disponga. Por el momento tan sólo existe configuración para los dispositivos Datalogic Blackjet™, que se corresponden con la configuración "PDA-WM5".



Pulsando sobre el botón de **Cambiar Configuración** podemos cambiar los parámetros de la configuración del dispositivo.

En el cuadro siguiente, en la pestaña de **Configuración de PDA** se selecciona el valor a dar a cada opción, existiendo las opciones de **Permitir**, **Preguntar** o **Prohibir**.

Hay tres apartados principales:

- **Sincronización de datos** : contiene opciones que determinan qué operaciones se realizarán durante una sincronización. Las operaciones que podemos especificar son : *Descargar lecturas*, *Descargar operarios*, *Descargar contadores* y *Descargar Rutas*. Para más información, consulte el apartado **Configuración de sincronización** del manual de [WATERLU](#).
- **Permisos** : establecen el comportamiento que tendrá el dispositivo ante situaciones anómalas. Las opciones chequeables establecen :
  - **Sobreescribir Configuración PDA** : si esta opción está activada, en la próxima sincronización se establecerá esta configuración en el dispositivo que se sincronice. Si no está marcada no se realizará ningún cambio en la configuración del dispositivo.
  - **Permitir Cambios Configuración Local PDA** : si esta opción está activada, en la próxima sincronización se inhabilitará el cambio de configuración localmente en la PDA. El servidor podrá seguir cambiando la configuración.
- **Alertas** : estas opciones establecen como reaccionará el dispositivo ante determinadas situaciones que se salen de lo habitual. Los valores posibles para estas opciones son **Permitir** (se procederá a realizar la acción solicitada, sin informar o preguntar), **Preguntar** (se pedirá confirmación para llevar a cabo la acción), o **Prohibir** (no se realizará la acción y se informará de ello).
  - **Lectura de Contadores Fuera de Ruta** : se da cuando al estar siguiendo una ruta se intenta medir un contador que, o bien no está en la ruta, o no es el siguiente contador a medir en la ruta. **Permitir** pasa a medir el contador directamente. **Preguntar** pedirá confirmación para

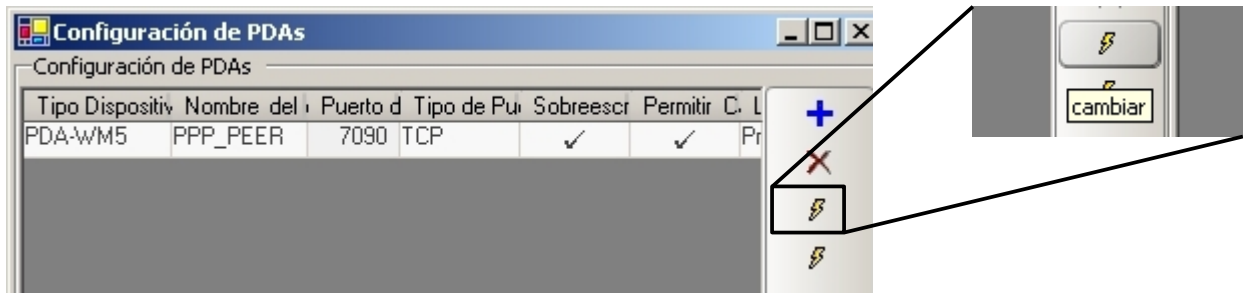
pasar a medir el contador fuera de ruta seleccionado. **Prohibir** no permitirá medir un contador que no sea el siguiente en la ruta y avisará de ello.

- **Relecturas** : se da al intentar tomar lectura de un contador que ya ha sido medido. **Permitir** pasa a medir el contador directamente. **Preguntar** pedirá confirmación para pasar a volver a medir el contador. **Prohibir** impedirá medir un contador que ya haya sido medido e informará de ello.
- **Lecturas Muy Bajas** : se da al introducir un valor de lectura inferior al rango considerado normal para ese contador. **Permitir** deja introducir valores bajos para la lectura. **Preguntar** pedirá confirmación para guardar la lectura introducida aún siendo baja. **Prohibir** no permitirá introducir lecturas con un valor bajo e informará de ello.
- **Lecturas Muy Altas** : se da al introducir un valor de lectura superior al rango considerado normal para ese contador. **Permitir** deja introducir valores altos para la lectura. **Preguntar** pedirá confirmación para guardar la lectura introducida aún siendo baja. **Prohibir** no permitirá introducir lecturas con un valor alto e informará de ello.
- **Preguntar en Caso de Incidencia** : se da al intentar introducir una incidencia para un contador. **Permitir** pasa directamente a la pantalla **Registro de incidencias**. **Preguntar** pedirá confirmación para pasar a la pantalla **Registro de incidencias**. **Prohibir** no permitirá introducir incidencias; se inhabilitará la pantalla **Registro de incidencias**.
- **Cambio de Ruta** : se da al tomar lectura del último contador de una ruta. **Permitir** hace que al medir el último contador de una ruta se salte directamente a la siguiente ruta. **Preguntar** pedirá confirmación para pasar a la siguiente ruta una vez medido el último contador de una ruta. **Prohibir** informará cuando acabe una ruta, pero no pasará a la siguiente.
- **Mostrar Contadores ya Leídos** : esta opción sólo admite **Permitir** o **Prohibir**. **Permitir** hace que se muestren los contadores ya leídos en la pantalla de **Seguimiento de ruta**. **Prohibir** evita que se muestren los contadores ya leídos en la pantalla **Seguimiento de ruta**.
- **Ir a Lectura tras Scan** : se da al leer un código de barras con el escáner del dispositivo. **Permitir** pasa directamente a medir el contador tras escanear el código de barras. **Preguntar** pedirá confirmación para pasar a medir el contador tras leer un código de barras. **Prohibir** hará que al leer un código de barras no se realice acción alguna.

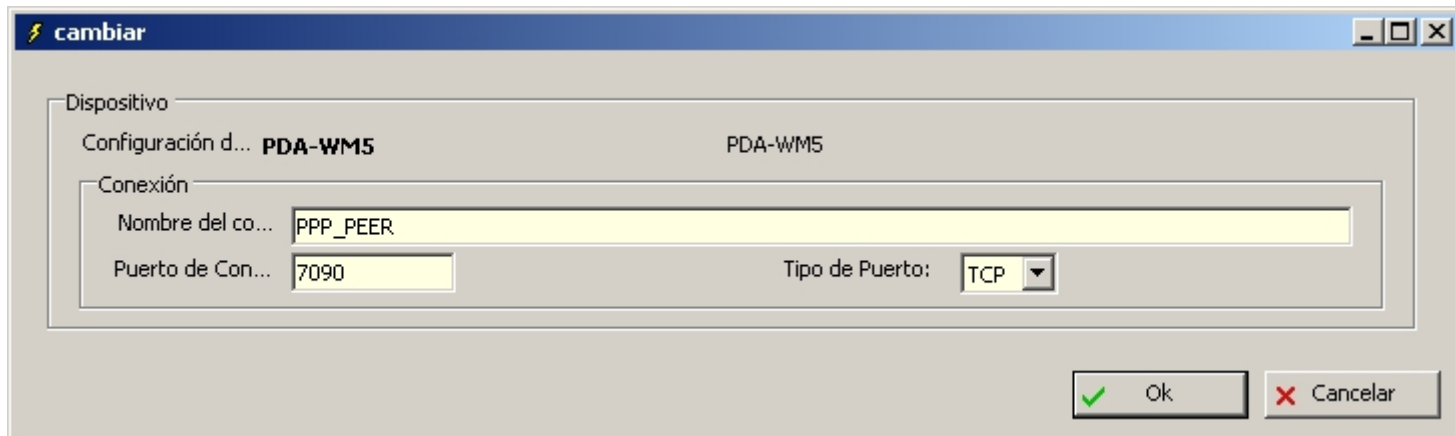
Cabe recordar que si la casilla de la opción **Sobreescribir Configuración PDA** está desactivada **no** se cambiará la configuración del dispositivo.

## **2.1 Configuración avanzada**

La configuración avanzada trata respecto del equipo, puerto y tipo de puerto a emplear durante las sincronizaciones. Llegamos a esta ventana mediante el botón **cambiar**.



Podemos cambiar la configuración de estos parámetros para cada tipo de configuración de que dispongamos. Los valores establecidos aquí se establecerán también en el dispositivo tras una sincronización si la casilla **Sobreescribir Configuración PDA** está activada.



Los parámetros de esta ventana indican :

- **Nombre del componente servidor** : el nombre DNS o dirección IP completa del equipo al que debe conectarse el dispositivo para realizar una sincronización. El valor por defecto es “PPP\_PEER”, el cuál es un alias que hace referencia al equipo que está conectado el dispositivo al emplear *ActiveSync*. Ejemplos de nombres válidos : “server.red.dominio”; “192.168.1.40”.
- **Puerto de conexión** : número de puerto en el que el equipo que sirve la sincronización está escuchando peticiones. El valor por defecto es 7090.
- **Tipo de puerto** : tipo de conexión a emplear durante la sincronización. Valores válidos son TCP ó UDP. El valor por defecto es TCP.

### 3 Impresión de códigos de barras

Una nueva funcionalidad de la aplicación es la posibilidad de imprimir el código de barras de un contador como un servicio más integrado en la aplicación.

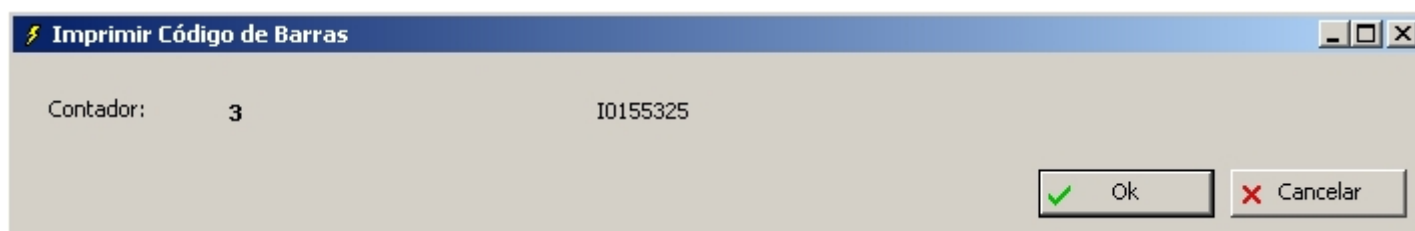
Para imprimir un código de barras no hay más que, o bien pulsar el botón **Imprimir código de barras** en un ventana de Contadores o sacar el menú contextual de un contador con el botón derecho y elegir la opción de **Imprimir código de barras**.



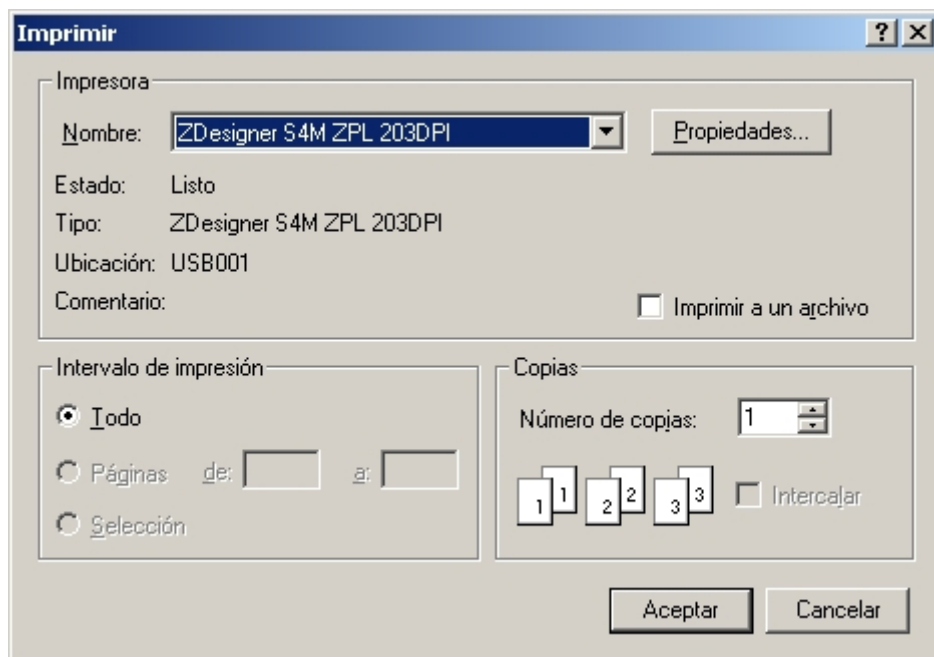
Si no había ningún contador seleccionado, se pide seleccionar uno de ellos.



Con un contador ya seleccionado

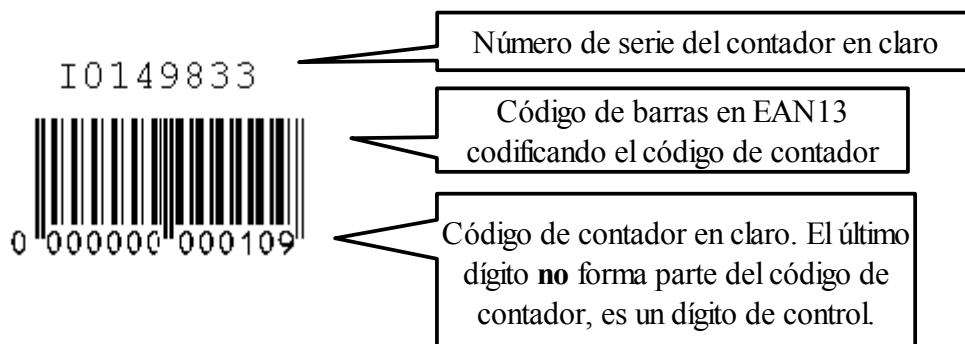


al pulsar **Ok** muestra el diálogo de configuración de la impresión, donde puede seleccionarse la impresora a emplear y configurar diferentes opciones de impresión según la impresora escogida.



Si están instaladas las impresoras **Zebra** en el equipo podrán emplearse para imprimir el código de barras sobre etiquetas plastificadas. Asegúrese de seleccionar la impresora cuyo nombre empiece por “**ZDesigner**”.

Una vez seleccionada la impresora y configurados los parámetros de impresión, pulsando aceptar se envía el código del contador a imprimir. Si todo funciona correctamente tendremos este resultado :



## 4 Impresión en Zebra

La impresión de un código de barras puede realizarse sobre cualquier impresora instalada en el equipo, pero ha sido especialmente diseñada para utilizarlo sobre la impresora de etiquetas **Zebra**.

Imprimiendo en esta impresora obtenemos el código de barras en un etiqueta preparada para su instalación en el contador. La etiqueta tiene el tamaño adecuado, ya está troquelada y su naturaleza plástica evita el deterioro por humedad o estancia en la intemperie.

### ***4.1 Instalación de la impresora Zebra***

Acompañando a la impresora hay un CD con documentación sobre el uso de la impresora así como los drivers necesarios para la instalación.

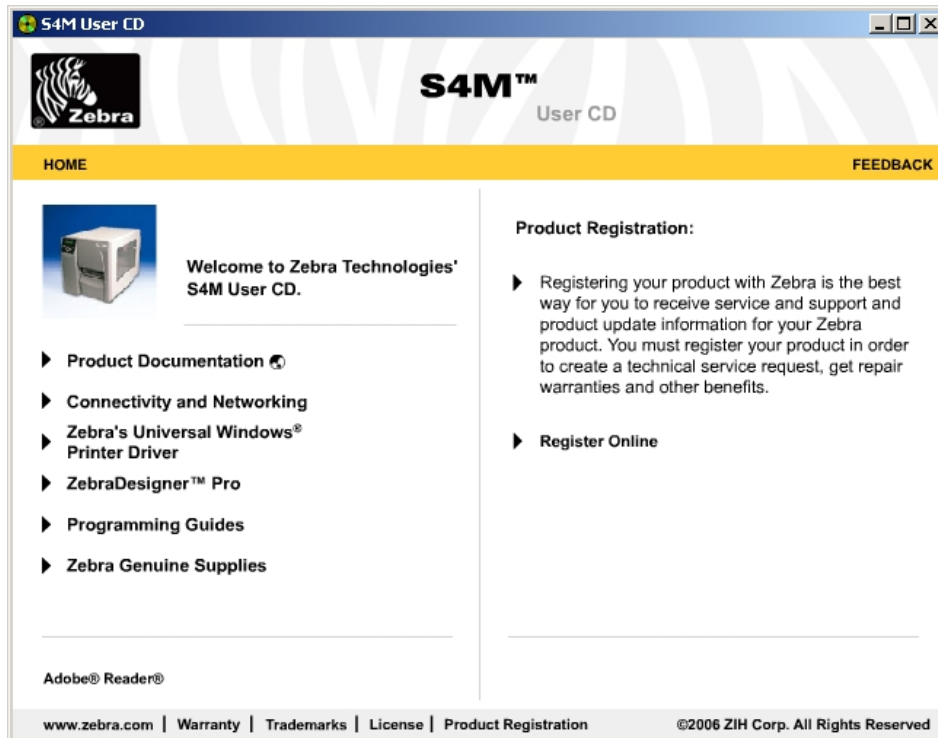
La instalación correcta de la impresora **Zebra S4M ZPL** 203 dpi implica varios pasos que deben realizarse en orden.

### 4.1.1 Instalación de Zebra S4M

El CD que acompaña a la impresora **Zebra S4M ZPL 203 dpi** incluye un programa de instalación que instala los drivers necesarios en **Windows**.

Para instalar la impresora siga estos pasos antes de conectar la impresora al equipo:

- 1) Introduzca el CD, y automáticamente se iniciará la pantalla de autoarranque.



- 2) Escoja en la izquierda la opción para instalar los drivers de la impresora para **Windows**:



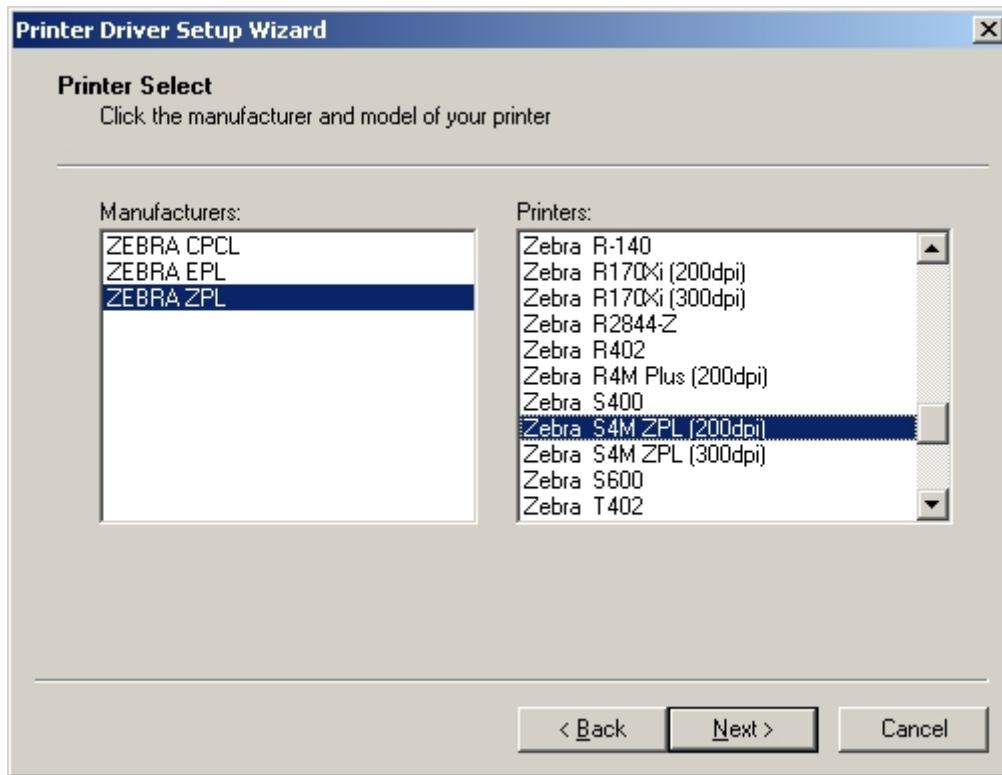
- 3) A continuación escoja la opción **AGREE** para instalar los drivers; o **VIEW** para ver la licencia de instalación. Puede pulsar **DISAGREE** si no está de acuerdo con la licencia, en tal caso no se instalarán los drivers.

#### License Agreement Terms and Conditions

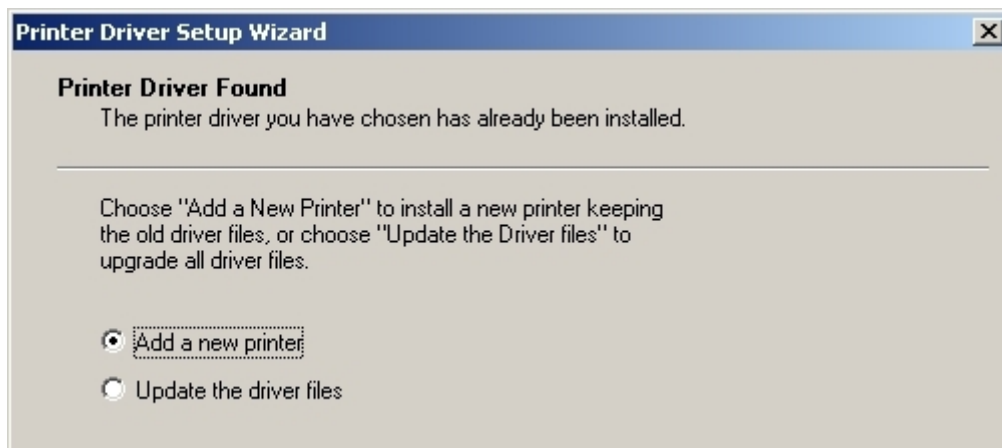
**VIEW**      **AGREE**      **DISAGREE**  
(Install Printer Driver)

- 4) Se inicia un proceso de instalación típico de **Windows**. Pulse **Next (Siguiente)** para continuar.
- 5) En la pantalla de selección del modelo de impresora, selecciones en la parte izquierda la opción **Zebra ZPL**, y en la derecha **Zebra S4M ZPL (200dpi)**.





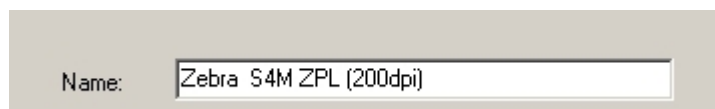
- 6) Si ya instaló la impresora previamente el instalador le preguntará si desea instalar una impresora nueva (**Add a new printer**) manteniendo la anterior o si desea actualizar los drivers (**Update the driver files.**)



- 7) Al añadir una nueva impresora se le preguntará por el puerto al que está conectada la impresora. Las opciones más comunes son **USB001** o **LPT1**.



- 8) El paso siguiente requiere que introduzca un nombre para la impresora. Puede dejar el valor por defecto.



- 9) Se copiarán los archivos necesarios en el sistema y terminará la instalación de los drivers de la impresora.
- 10) Conecte ahora la impresora al equipo. **Windows** reconocerá el nuevo dispositivo e iniciará el asistente de instalación de dispositivos **Plug & Play**.
- 11) Al iniciarse el asistente le preguntará si desea conectarse a **Windows Update** para instalar el nuevo dispositivo. Elija la opción **“No por el momento”**.

¿Desea que Windows se conecte a Windows Update para buscar software?

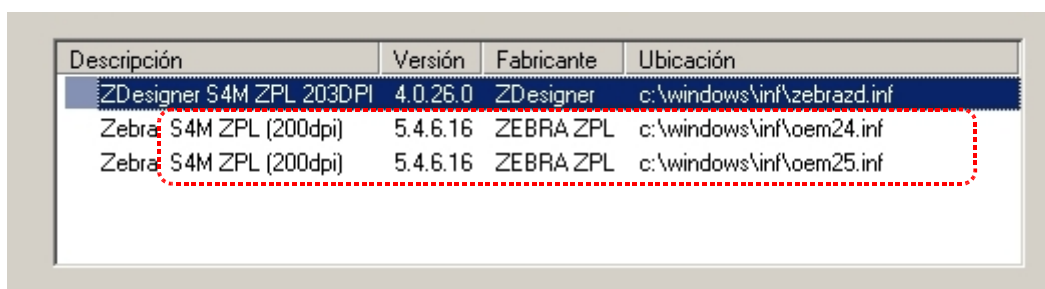
- ☐ Sí, sólo esta vez
- ☐ Sí, ahora y cada vez que conecte un dispositivo
- ☒ No por el momento

- 12) Elija la opción **“Instalar automáticamente el software (recomendado)”**. Si el CD que acompaña a la impresora no está introducido, introdúzcalo antes de pulsar siguiente.

¿Qué desea que haga el asistente?

- ☒ Instalar automáticamente el software (recomendado)
- ☐ Instalar desde una lista o ubicación específica (avanzado)

- 13) El asistente le mostrará una lista de dispositivos. Seleccione el dispositivo **Zebra S4M ZPL (200dpi)**. Si aparece más de uno con ese nombre, cualquiera de ellos es válido.



- 14) Si el asistente muestra una ventana de aviso informando de que el software a instalar no ha pasado la prueba del logotipo de **Windows** pulse **Continuar**.



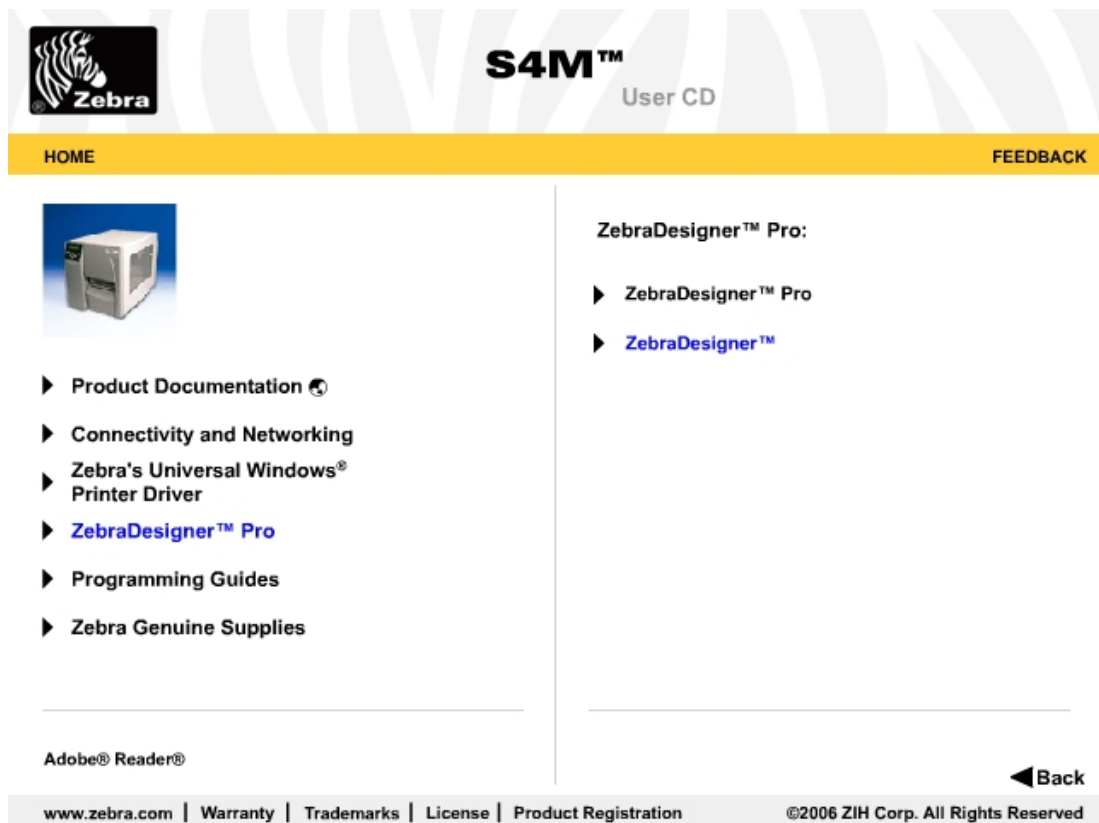
- 15) Se copiarán los archivos necesarios y la impresora estará instalada y reconocida por **Windows**.

### 4.1.2 Instalación de ZebraDesigner

Es necesario instalar el programa **ZebraDesigner** que viene en el CD que acompaña a la impresora. Este programa nos permitirá instalar una impresora virtual encima de la impresora **Zebra S4M ZPL 200dpi** que será la que emplearemos realmente para imprimir.

Para instalar la aplicación **ZebraDesigner**, siga estos pasos:

- 1) Introduzca el CD, y en el menú de autoarranque seleccione en la parte izquierda la opción **ZebraDesigner Pro**. Tras hacer clic sobre esa opción, seleccione a la derecha la opción **ZebraDesigner**.



- 2) Seleccione ahora la opción “**Install ZebraDesigner**”

- ▶ **Install ZebraDesigner™**
- ▶ ZebraDesigner™  
Documentation & Manuals

- 3) Se iniciará un asistente de instalación típico de **Windows**. Seleccione las opciones que desee y al terminar el programa **ZebraDesigner** estará instalado en el equipo.

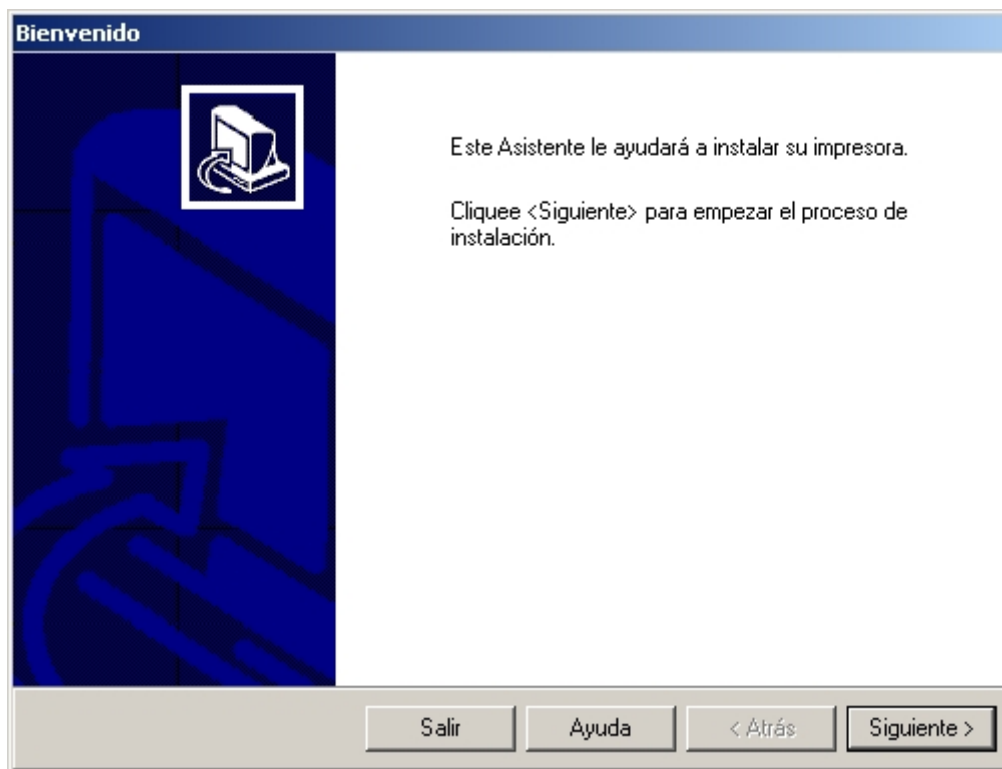
### 4.1.3 Instalación de impresora ZDesigner

Instalaremos ahora una impresora virtual encima de la impresora **Zebra S4M ZPL 200dpi**.

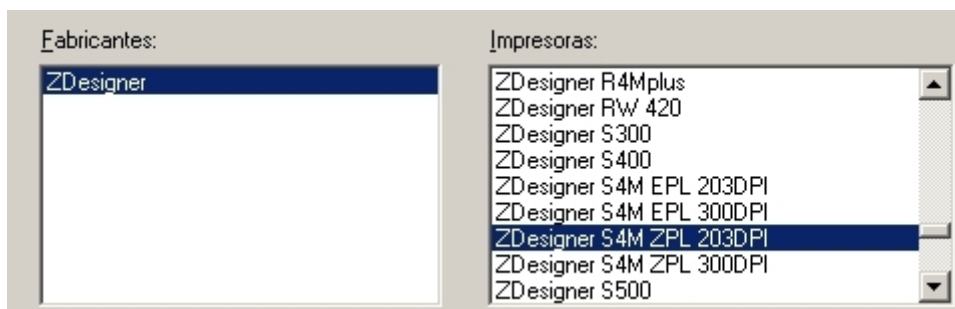
- 1) En el menú “**Inicio**” seleccione la carpeta del programa instalado anteriormente **ZebraDesigner**, y dentro de las opciones que aparecen seleccione “**Add Printer**”.



- 2) Se iniciará un asistente para instalar la impresora virtual.

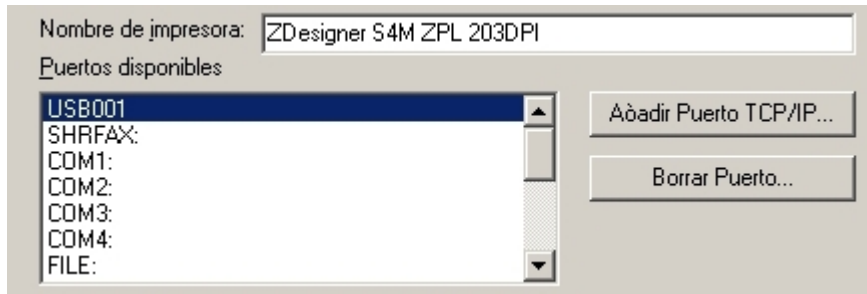


- 3) En la lista, seleccione a la izquierda la opción **ZDesigner**, y a la derecha **ZDesigner S4M ZPL 203 dpi**.



- 4) Introduzca (o deje el valor por defecto) un nombre para esta impresora virtual, y seleccione el puerto que empleará. Para obtener una buena integración con la impresión desde **AguasCli** es **importante** que el nombre

de esta impresora comience por “**ZDesigner**” (las mayúsculas son importantes), y que el puerto escogido sea el mismo que el de la impresora **Zebra S4M** real.



- 5) Se copiarán los archivos necesarios en el sistema.



- 6) Una vez terminen de copiarse los archivos, la impresora virtual estará instalada.

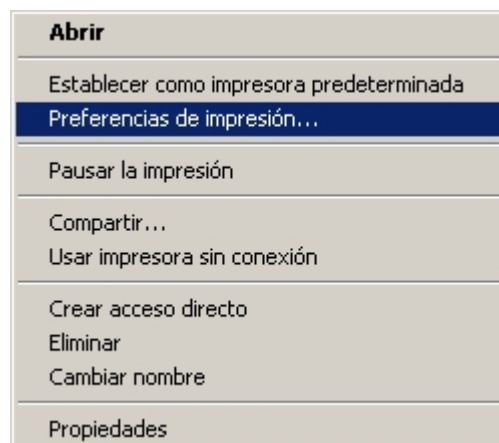
#### 4.1.4 Configuración de ZDesigner

Una vez instaladas la impresora real y la virtual es hora de configurarlas para la impresión de las etiquetas.

- 1) Deben aparecer ahora las dos impresoras instaladas en el apartado “Impresoras y Faxes” del “Panel de control”.



- 2) Haga clic con el botón derecho sobre la impresora virtual. En el menú que aparece seleccione “Preferencias de impresión...”.



- 3) En la ventana con pestañas de configuración seleccione la pestaña “Opciones”. En ella ponga los valores como los mostrados en la imagen:



- 4) Pase ahora a la pestaña “**Configuración avanzada**” y ponga los valores igual que los mostrados :

Opciones Configuración Avanzada Interpolación Stocks (almacenes)

Modo operación

☒ Desplazar ☐ Cortador

☐ Rebobinar Cantidad de 0

☐ Pelar ☐ Avanzado<sup>xxx</sup>

☐ Aplicador ☐ Imágen de espejo

Tipo de media

☒ Transferencia termal ☐ Termo-Directo

Modo Tracking

☐ Continuo ☒ Sensor de marca

☐ Web Sensing

Ajuste

Arriba 0,00 cm

Desplazar 0,00 cm

El campo **Arriba** deberá ser ajustado tras haber realizado alguna prueba de impresión. Dependiendo del desplazamiento que muestre deberá establecerse el desplazamiento. Experimente con esta opción hasta hallar el valor adecuado.

- 5) Las impresoras están listas para imprimir.



- Página intencionadamente en blanco -



